INSIDE LINUX Learn what makes it tick and how to use it

FAIRPHONE 5 Discover the ethical hardware business

RETRO GAMING Create the ultimate old-school experience

LINUX FORMAT

The #1 open source mag

TEXT EDITORS Five powerful apps for managing big projects

UPGRADE IT!

THE ULTIMATE DESKTOP

Enhance your Linux desktop with the most advanced KDE framework ever!

100 pages of Linux tricks, tips & more!

PLUS: HOW TO

- Code classic Amiga-style demos
- Recreate the original 1970s Microtan
- Design a modern play-by-mail game

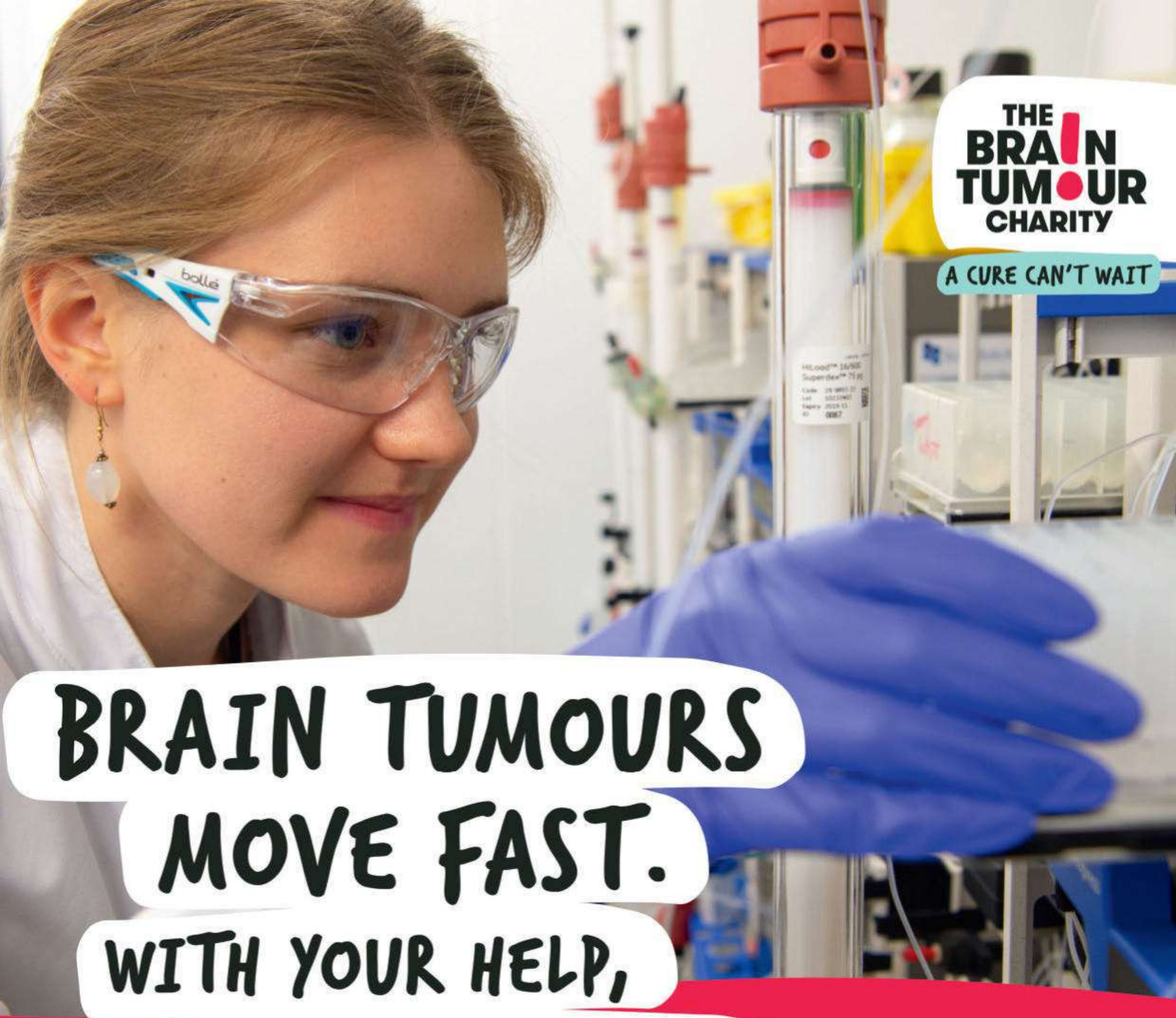
WEATHER Pi Build your own always-on rain detector

PRIVATE CALLS Secure your video chat with open source

GRAPHICS TABLETS Get to grips with Krita's pressure-based brushes

LXF July 2024

FUTURE



We're working to create a future where brain tumours are curable.
We urgently need your help to accelerate research.

Text DEFEAT5 to 70507 to donate £5, please help us to find a cure.

thebraintumourcharity.org

© The Brain Tumour Charity 2020. Registered Charity in England and Wales (1150054) and Scotland (SC045081)

FR Registered with FUNDRAISING REGULATOR

LINUX FORMAT

Corporate Patron Free Software Foundation

» MEET THE TEAM

We're looking at the KDE Plasma 6 desktop environment this issue, but what's your favourite desktop of all time?

Jonni Bidwell
I'll always have a soft spot for Enlightenment, as used in Bodhi. It seemed so futuristic in 2006, and provided endless things to configure when I had lots of other stuff I should've been getting on with. Tiling window managers and tmux-ed terminals are timeless, though.

Michael Reed
I was a fan of KDE 3 as the technical user's system and Gnome 2 as the business and education desktop. Credit where credit's due, the Windows 95 layout with the task bar and app launcher on the bottom (which they both 'embraced') is an efficient system.

Nick Peers
Mine is Ubuntu's flavour of Gnome. It is my day-to-day desktop, after all. However, I'm also keeping a keen eye on Pantheon, Elementary OS's desktop environment. It's modern, sleek and slowly evolving into a true work of art that's both practical to use and beautiful to behold.

Les Pounder
I've loved Openbox since I first used CrunchBang Linux (#!), over a decade ago. The light system requirements and dark theme suit my old eyes and laptop. These days I use Crunchbang++ or BunsenLabs, which both carry on the traditions and aesthetic set by #!

Mayank Sharma
Truth be told, I only think of the desktop when my old workhorse huffs and puffs with the factory-fitted DEs of the latest distros. For them, I assemble my own atop Openbox, with *PCManFM* and Tint2. If I am feeling lazy, I'll just use LXDE and be done with it.

*Savings are based on the cover price.

Desktop upgrades

Everyone thinks they know best when it comes to desktops. Having just looked at 20 years of Ubuntu, it was funny to recall users complaining when the Unity desktop was introduced and then users complaining when the Unity desktop was retired – for what was a mostly identical Gnome desktop experience. From Cinnamon to Budgie to Mate and Xfce, the range of open source desktops is wide and well supported. So, this issue we're taking a good look at the latest KDE Plasma 6, something we haven't done in far too long and arguably the most powerful desktop around. I'm sure you'll tell us if you disagree!

It's an ideal time, too, because not only has the KDE project updated the Plasma desktop, but it's also rolling out the new KDE Framework 6 and KDE Gear 24 alongside it, partly in response to its GUI toolkit *Qt 6* being overhauled and released. As always with such a large release, there have been bugs to be squished – and they have been; by the time you're reading this, KDE Plasma 6.0.5 should be out.

But the beauty of open source – if you don't like it, go try something else! We're kicking off a new Inside Linux series looking at the different parts of a Linux system, starting with the various kernel subsystems and how drivers work within Linux and how you can control them. It's all about giving you, the user, more control – if you want it.

Neil

Neil Mohr Editor
neil.mohr@futurenet.com

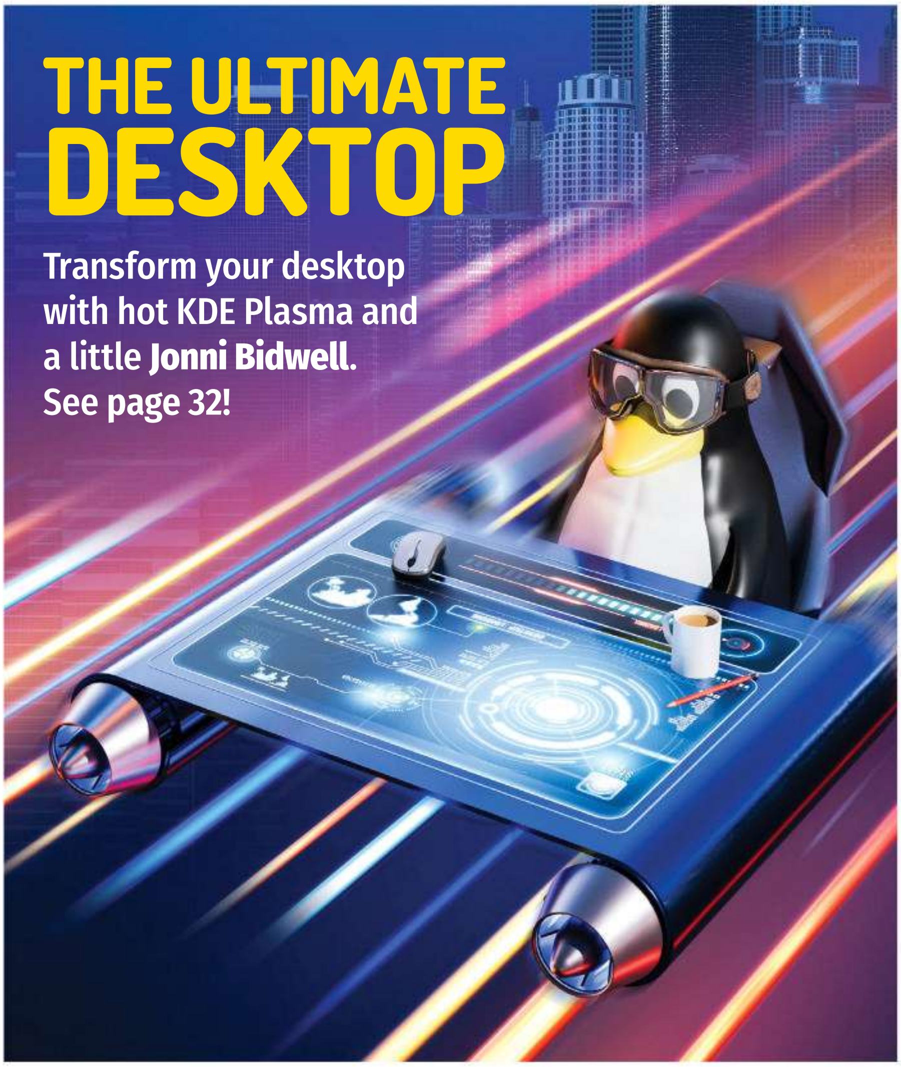
SAVE! 50%*

Subscribe & save!
On digital and print: see page 16

Contents

SUBSCRIBE NOW!
Page 16

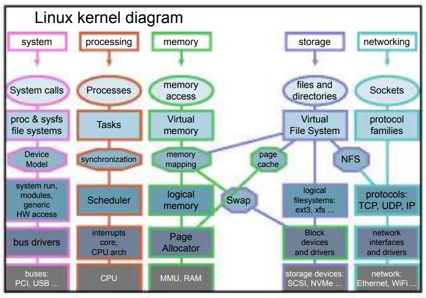
REVIEWS



CREDIT: Magictorch

ROUNDUP

IN DEPTH



CREDIT: Getty Images/Sean Gladwell, https://github.com/makelinux

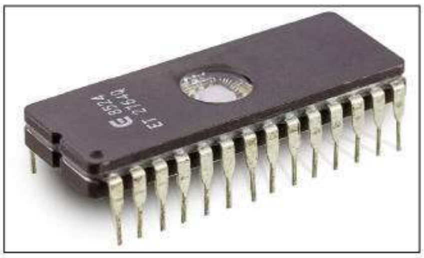
CONTENTS

Pi USER

CODING ACADEMY

REGULARS AT A GLANCE

TUTORIALS



ADMINISTERIA

Newsdesk

THIS ISSUE: Lightning Linux » Zombie Z80 » Mutilated MS-DOS » Sudo substitute » Deutsch docs go open source

OPERATING SYSTEMS

Ubuntu 24.04 scores 20% faster than Windows 11

Over 100 benchmarks on the AMD Ryzen 7 7840HS Framework 16 have shown that Ubuntu 24.04 seriously outperforms Windows 11.

Benchmarks performed on Ubuntu 23.10, Ubuntu 24.04 and Windows 11 have come down strongly in favour of Noble Numbat. The results, which can be viewed at **https://openbenchmarking.org**, are the result of 104 different tests. In over two thirds of these (67%), Ubuntu 24.04 came out fastest. Ubuntu 23.10 was a respectable silver medallist (22%) and Windows 11 took the bronze at just 10%.

Given how entrenched Linux and Windows fans can be, these statistics have provoked controversy in online discussion groups.

Much of this is around the claim that Ubuntu 24.04 is 20% faster than Windows 11. The figure was obtained by analysing the geometric mean of all results, showing that Ubuntu 23.10 was 16% faster on average than Windows 11. Ubuntu 24.04 was around 3% faster again, hence the claim that the OS is 20% faster than Windows on an AMD Ryzen 7 7840HS laptop.

In the case of each OS, the benchmarks were run on a clean install, using default settings. During the tests, the laptop was connected to AC power and set to balanced mode in the OS.

The choice of the Framework 16 laptop is surprising, given that the series has drawn criticism from tech websites such as Arstechnica for a relatively slow software and firmware update process, as well as the long-awaited Linux-based BIOS updater, which at the time of writing, still isn't available.

Online commenters have also expressed surprise that the results are so divergent, given that many of the tests are CPU-based. This might be explained by AMD's contributions to the Linux kernel, as well as open source drivers for various CPUs and GPUs.

According to the April 2024 Steam Hardware and Software Survey, the speed of AMD CPUs at 3.7GHz and above has improved 1.15% month on month, versus 0.84% for Windows. This may explain why Steam reports that AMD CPU use by Linux gamers is now over 71%.

Much of this can be accounted for by the Steam Deck, which uses a custom AMD CPU. It also may explain the wide disparity in the Windows/Ubuntu benchmarks.

For instance, playing games through the Steam Deck's Proton compatibility layer should mean they underperform relative to running on Windows. In practice, there's much anecdotal evidence claiming games run better via Proton.

Most Linux lovers put this down to the unnecessary number of background processes in Windows, which can reduce performance.

The benchmarks were carried out on a Framework 16 laptop. The manufacturers are still working on full Linux compatibility.

CREDIT: Framework

KERNEL BLIP

The divergent results might be explained by AMD's contributions to the Linux kernel, as well as open source drives.

HARDWARE

Open source Z80 alternative

Zilog has announced the EOL of the legendary Z80, but it may live again.

Although the Z80 has officially reached its EOL, it may live on through FOSS Z80, an open source clone.

On April 15th, Zilog announced that after "just" 48 years, production of certain models of the Z80 will end. The 8-bit processor could be found in any number of mainstream products throughout the '70s and '80s, including the ZX Spectrum and Sega Mega Drive, as well as many early synthesisers, graphing calculators and portable computers.

The chip was originally designed for binary compatibility with the Intel 8080, and initial marketing was focused on this, as well as the advantages it offered. Intel ultimately rebounded with a series of rival processors, causing Zilog to focus mainly on microcontrollers instead.

Given the popularity of the chip and its derivatives, it's hardly surprising that the open source and hardware preservation community decided to step in, through the FOSS Z80.

Founded by Unity iOS developer Renaldas Zioma, the project's GitHub page's stated goal is to develop a silicon-proven, pin-compatible, open source replacement for classic Zilog Z80. The chip will be based on Guy Hutchison's TV80 soft core and has a tentative release date of June 2024.

Given the Z80's age and the number of clones that have previously been produced, such as the Soviet KP1858BM1, reverse engineering the chip won't be an issue. However, the challenge remains of producing physical samples for testing purposes without spending tens of thousands of dollars.

Zioma plans to do this expediently through using Tiny Tapeout. This MPW (multi-project wafer) service uses Skywater's open source 130nm PDK on a 0.064mm^2 die area. This can hugely cut down production costs by fitting multiple die designs on to the same silicon wafer.

Read more at **https://github.com/rejunity/z80-open-silicon**.

OPERATING SYSTEMS

Microsoft open sources MS-DOS

Criticism has been drawn for the "mutilated" release.

On 25th April, Microsoft announced that, in partnership with IBM, it has released MS-DOS 4.00 under the open source MIT licence.

This release of the OS, which originally came out in 1988, includes its source code, binaries, disk images and documentation.

The impetus came from British researcher Connor 'Starfrost' Hyde, who contacted CTO Ray Ozzie about his software collection.

While working for Lotus, Starfrost had received some unreleased beta binaries of DOS 4.0 on floppy disk. Volunteers imaged these and digitised the paper manuals ahead of the project release on GitHub.

The following day, Michal Necasek, operator of the popular OS/2 Museum blog, posted an article in response entitled 'How Not to Release Historic Source Code'.

Necasek acknowledged it's "brilliant" that Microsoft has made MS-DOS 4.00's source code available, but noted that the Git upload process broke most timestamps, as well as encoding virtually all files to UTF-8. The result was to "mutilate historic source code", making it difficult to run without extensive debugging.

In the comments, Necasek grants that Git doesn't keep timestamps by design, but goes on to claim that's what makes GitHub unsuitable for uploading historic source code of this kind.

He concludes that such code could easily be preserved and shared simply by releasing it as an archive of files in ZIP or tar format. We at **LXF** are inclined to agree.

CREDIT: Wikimedia/Yaca2671, https://commons.wikimedia.org/wiki/File:Z0840008PSC_01.jpg

OPINION

YOU NEED HELP!

Italo Vignoli is one of the founders of LibreOffice and the Document Foundation.

LibreOffice 24.2 Writer and *Calc* manuals are available for free as standard ODT and PDF files from **https://books.libreoffice.org.** They have been edited by a small group of volunteer contributors: Jean Hollis Weber for *Writer,* and Steve Fanning and Olivier Hallot for *Calc.*

Previous versions of the guides, on which the updated versions are based, were edited by other volunteers: Antonio Fernández, Dave Barton, Gordon Bates, Bruce Byfield, Kees Kriek, Rafael Lima, Skip Masonsmith, Leo Moons, Kieran Peckett, Gillian Polack, Peter Schofield, Felipe Viggiano and Zbyszek Zak. Anyone who needs to get up to speed with the new versions of *Writer* and *Calc* will find these two books invaluable.

The guides are an addition to *LibreOffice's* extensive online Help system. Windows and Linux users can choose to download and install the offline Help for use when not online, while the offline Help is installed together with the program on Mac OS.

To display the Help system, press F1 or select Help > LibreOffice Help from the menu bar. If you do not have the offline Help installed but are connected to the internet, the default browser opens the online Help pages on the *LibreOffice* website.

OPINION

IN THE CLOUDS

David Stokes, is a technology evangelist at Percona.

“On 6th June, it will be the tenth anniversary of the launch of *Kubernetes,* the container orchestration tool developed by Google. It was developed to make it easier to deploy software in a repeatable and predictable manner using container images, then manage those workloads over time.

Since launch, *Kubernetes* has grown from a nascent technology useful for very geeky workloads into a platform that is used every day. It is deeply complex and technical, but aims to get the most out of cloud computing deployments by automating common steps and making it easier to run in the cloud.

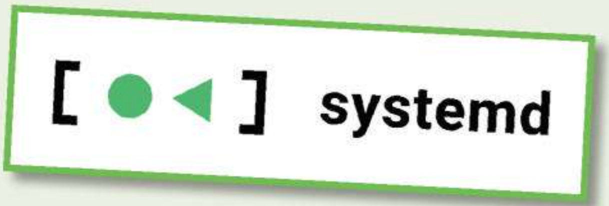
One important element in its success is that it is open source, given over to the Linux Foundation under the Apache License as the bedrock for the Cloud Native Computing Foundation. The most valuable thing *Kubernetes* did was make it easier to shift cloud workloads.

We shouldn’t ignore Google’s reasons for this – cloud portability is useful when competing against the likes of AWS and Microsoft – but *Kubernetes* being open source made it possible for all cloud providers to offer services and get involved. We are still in the development stage around cloud-native infrastructure, but *Kubernetes* got us started.”

PRIVILEGES

Systemd wants to add sudo replacement

The upcoming v256 release of Systemd is expected to include a sudo replacement called run0.



Actually an implementation of the existing systemd-run, run0 has a much smaller attack surface than sudo.

CREDIT: Wikimedia

In early May, *Systemd* lead developer Lennart Poettering posted the fifth in a series of posts on key new features of the upcoming v256 release. He began by acknowledging the “venerable” *sudo* tool in that, in theory, it allows ordinary users to operate at minimum privilege, unless elevated privileges are required for specific tasks.

However, Poettering also grants that *sudo* is a relatively large SUID binary with a complicated configuration language, loadable plugins and hostname matches. While acknowledging the existence of viable alternatives like OpenBSD’s *doas*, which greatly reduce the attack surface, such tools are still SUID binary. In Poettering’s view, this makes them inherently inefficient given that SUID processes must inherit the execution context intended for and controlled by unprivileged code such as environment variables and process scheduling.

When invoked, *run0* will perform the same function as *sudo* but not as an SUID. Instead, it requests the service manager to invoke a command or shell under the target user’s UID.

SOFTWARE

German state moves to LibreOffice

Schleswig-Holstein switches from Microsoft Office.

After a successful trial, Germany’s northernmost state has decided that all 30,000 local government PCs will move away from Windows and *Microsoft Office* in favour of Linux. Use of *LibreOffice* will be mandatory and staff training will be provided.

Minister President Daniel Gunther announced that Schleswig-Holstein will be “the first state to introduce a digitally sovereign IT workplace in its state administration”.

He went on to announce adoption of other open source products on state computers, including *Nextcloud* and *Thunderbird.*

State officials and the Document Foundation point out that LibreOffice will give the state much better control of user data.

MALWARE

Compromised Dockers

Docker Hub hosted millions of imageless containers.

Research by JFrog has uncovered coordinated attacks on Docker Hub that planted millions of malicious repositories. Docker Hub allows repository maintainers to add descriptions and documentation in HTML to a repo’s main page.

JFrog security researcher Andrey Polkovnichenko discovered that around 4.6 million such repositories contained no content except for said overview page, which held links to malware.

Docker has since taken these pages down but as Shachar Menashe of JFrog Security notes, “There’s not a lot that users can do to protect themselves at the outset, other than exercising caution.”

ollama/ollama

Docker Pull Command

Ollama Docker image

CPU only

Bad actors can create bogus overview pages with malicious links.

CREDIT: Wikimedia, Docker Hub

Distro watch

What's behind the free software sofa?

RED HAT 9.4

Red Hat 9.4 is now available on x86, ARM, IBM Power, IBM Z and IBM LinuxONE platforms to existing customers. New customers can also benefit from a free 60-day trial. The *Image Builder* has received an overhaul. Users can now specify arbitrary custom mount points, with the exception of those reserved for the system itself. The OS also has upgraded security, with support for passwordless biometric authentication and SELinux 3.6. Read more at **www.redhat.com**.

Red Hat 9.4 comes with heavy security upgrades.

LIBREELEC 12.0

LibreELEC is an independent fork of OpenELEC. The project website describes it as "just enough OS to run *Kodi* media centre". The latest version (dubbed Omega) has changed all 64-bit capable ARM system-on-a-chip devices, including the Raspberry Pi 4 and 5, from ARM to AArch64 userspace. The Linux kernel itself has been updated to version 6.6.28. Updating from LibreELEC 9.x and older iterations requires a clean install due to the Python 3 changes. You can learn more at **https://libreelec.tv**.

LibreELEC is great for running the Kodi media centre.

PRIMTUX 8

The first release of Primtux (code name Eiffel) was created in 2016 by a team of French educational enthusiasts and programmers. The OS isn't designed for personal use so much as a drop-in replacement for Windows on ageing PCs. The latest version is based on Linux Mint 21.3 and contains a new menu and search bar for exploring educational apps. The site's documentation has also had an extensive rewrite. French speakers can read more at **https://primtux.fr**.

Primtux is a French distro for educational environments.

GARUDA LINUX 240428

This rolling distribution is based on Arch Linux but has the benefit of a graphical installer *(Calamares).* Garuda itself is primarily designed for performance optimisation, so comes with a number of tweaks to this end, such as using zram and a CPU governor. Highlights of the latest release (code name Bird of Prey) include an upgrade to the Plasma 6 desktop environment as well as major improvements to *Sway* and *Hyprland.* You can discover more by pointing your browser at **https://garudalinux.org**.

Garuda contains tools for optimal performance.

RISC OS 5.3.0

RISC OS was designed by Acorn Computers in 1987 to run on the ARM chipset. Given the explosion in popularity of ARM boards, it's hardly surprising that the OS has had a new lease of life. The latest version is compatible with Beagleboard, IOMD, Iyonix, OMAP5, Pandaboard, Raspberry Pi and Titanium. RISC OS also includes "a bumper crop of 347 improvements to the HardDisc4 image and applications, and 329 for the main ROM image". Learn more at **www.riscosopen.org**.

RISC OS is compatible with ARM boards such as the Raspberry Pi.

OPINION

KEEPING IT REAL

Frederic Plourde
is XR lead at Collabora.

"April 2024 marked a milestone in the realm of extended reality (XR) development as the Khronos Group unveiled the highly anticipated OpenXR 1.1 specification. It signifies an important leap forward by consolidating multiple OpenXR extensions into the core spec. Users can enjoy simplified programming and streamlined cross-platform development processes for XR applications.

The integration of several extensions into the core specification not only fosters wider adoption but also provides developers and vendors with a unified foundation, ensuring consistent and seamless cross-platform user experiences. Notable extensions promoted to the core spec include XR_EXT_local_floor, XR_VARJO_quad_views, XR_EXT_palm_pose, XR_EXT_uuid, and XR_KHR_locate_spaces.

Monado, the cross-platform open source XR runtime, has received significant updates to align with the features and specs of OpenXR 1.1, ensuring compatibility and optimal performance for developers and users alike. By staying on par with OpenXR 1.1, Monado continues to be a robust and versatile platform for creating immersive free and open source XR experiences across a wide range of devices and software environments."

OPINION

POWER DOWN

Jon Masters is a kernel hacker who's been involved with Linux for over 22 years.

"A recent patch series removed support for the IBM PowerPC 405. If you worked on embedded Linux devices around the turn of the century, you will have encountered many of these. They were (briefly) everywhere back in the twilight years of the PowerPC architecture. This was a time before Apple had moved to x86 (and ultimately, Arm), when PowerBooks were called such because they had PowerPC processors inside. I was a huge fan of the PowerPC architecture at the time, since it was RISC-based and used in my first big commercial embedded Linux projects – a piece of scientific instrumentation.

Also, Arnd Bergmann posted patches that remove support for older Alpha processors (EV5 and earlier). Ostensibly because EV5 is the last processor in the kernel that can't support native byte access, one can't help but think that the days of Alpha support are also numbered.

People are entering the industry who were born after both of these CPUs had long gone out of fashion. Linux removes support for older hardware all the time. But it does feel as if we are entering a new era, one in which the variety of the past gives way to three main choices: Arm, RISC-V and x86. This isn't a bad thing per se, just a change."

Kernel Watch

Jon Masters keeps up with all the latest happenings in the Linux kernel, so you don't have to.

Linus Torvalds has announced Linux 6.9 release candidate (RC) 7. It is rare to go beyond this point in a development cycle, and thus likely that 6.9 will be released imminently. Probably the most interesting stuff in this release is under the hood, but a few things stand out as of interest to end users. This includes support for AMD SNP (Secure Nested Paging) Confidential Compute guest VMs. It may also be interesting to note that Linux 6.9 will officially deprecate the venerable ext2 filesystem driver. Support for older filesystems will exist most likely for decades – many people have used ext2 on SD storage to easily avoid journaling wear issues, though this can be disabled in ext4 – but users are being encouraged to avoid ext2 because it cannot represent timestamps beyond those supported by 32-bit Unix timestamps (the Y2k38 problem, which you can read about at length in our previous issue, **LXF315**).

IT'S LINUX, OF COURSE!

"Which is exactly what Apple did in a special undocumented feature only made available to Rosetta on Mac OS. Except that (of course) Linux can run on these Macs."

TSO memory model

Modern computers are usually formed from a large number of individual CPU cores that operate upon shared memory visible to all of them. As users of these machines, we demand the highest level of performance. But there is a tension between performance and certain intrinsic behaviours inherent to hardware, such as the communication mechanics of data sharing across large numbers of cores. In order to increase performance, certain 'relaxations' are often introduced into hardware designs that can (for those not expecting it) be surprising.

A classical example comes in the form of two CPU cores that both operate upon two shared data items (variables), located in memory locations x and y (which start out containing the value zero). Processor A writes the value 1 into location x, then reads y into register (memory local to the core) r1. Meanwhile, processor B writes the value 1 into y, then reads x into its register r1. You might expect that one (or both) of the two processors will write a 1 first, and that either (or both) will see a 1 in its register r1. But in truth, thanks to a hardware feature called Store Buffering (SB), it is possible for both cores to see the value zero (because the store of 1 on each core is written into a local store buffer not observed by the other core).

To make life easier for system designers and low-level software programmers, permitted behaviours and relaxations (known in the industry as memory consistency models) are described for each architecture. Some are strong (limit visible optimisation), while others are relaxed, requiring a programmer to insert special 'barrier' instructions to maintain relative ordering of memory operations observed by other processors.

x86 is traditionally strong and Arm is relaxed. This can result in porting challenges while running x86 code under emulation on Arm, and can require expensive memory barriers to preserve correct program behaviour.

Total Store Order (TSO) is the model used by x86. While, generally, relaxed hardware designs (as used by Apple Silicon) can lead to higher performance, the cost of emulating TSO in software is quite high relative to adding special hardware support. Which is exactly what Apple did in a special undocumented feature only made available to Rosetta on Mac OS. Except that (of course) Linux can run on these Macs, and folks have figured out how to enable TSO more broadly. But there is pushback against generally making this feature available to all users, since it can have a tendency to paper over software bugs if developers can 'just turn on TSO'. And with a significant impact on those platforms that don't have TSO support.

Answers

Got a burning question about open source or the kernel? Whatever your level, email it to **answers@linuxformat.com**

Neil Bothwick believes that he is the reincarnation of Alan Turing.

Q Hotting up Debian

I've started using Linux through Debian and so far so good. However, something that's pretty essential to how I use my computer is the various shortcuts *AutoHotKey* provided in Windows.

In *AutoHotKey,* I had a script that played an MP3, located in my files, every 20 minutes, which was an alarm notice for me; I had various text replacement tools that enabled me to enter long strings of text wherever I happened to be typing by just typing an arbitrary code phrase and pressing Space; and I had macros that would do things like run a Java file from the command prompt just by pressing F9.
Evie Burke

A Welcome to the world of Linux. One of the many differences between Linux and Windows is that there is usually more than one way of doing something in Linux. This includes fundamentals such as the desktop environment. You do not say which desktop you are using, but many have the ability to respond to hotkeys built in. If you go into your desktop's main configuration window, you will almost certainly find such an option that allows you to bind actions to keys. Those actions could be as simple as inserting some text or could run a command.

If you are looking for a method independent of your desktop, there is *Autokey* (**https://github.com/autokey/autokey**). Although we have given the homepage here, that is not how you install software in Linux; instead you do it from the distro's software manager, usually *Synaptic* on Debian. *Autokey* does most of what you describe and the wiki linked from the homepage gives clear instructions.

One thing that you do with *AutoHotKey* that is not covered by *Autokey* is run a command at regular intervals. One of the mantras of the Unix/Linux world is that each tool should do one thing and do it well. Responding to hotkey events is handled by *Autokey*, or the desktop's own equivalent; running scheduled tasks is handled by *Cron,* software designed to do just this. *Cron* is controlled by files called crontabs, there is one to run system commands, plus each user can have their own. You create or edit a crontab file with:

```
$ crontab -e
```

or

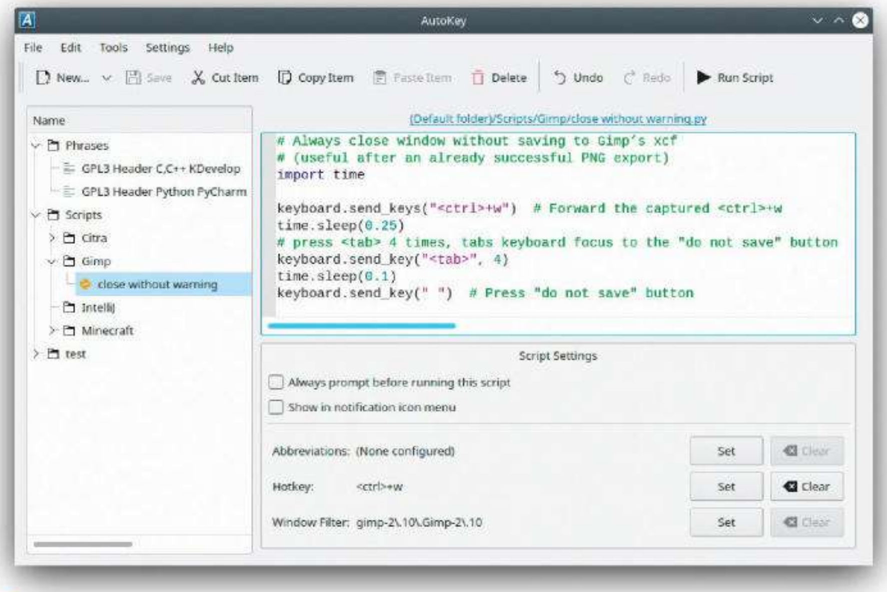
```
$ sudo crontab -e -u username
```

The first method is preferred but some distros are set up to only allow the root user to create crontabs, so we have to use *sudo*. This opens a crontab containing only explanatory comments, to which you can add a line to define each command. Each line contains six items, the first five are the minute, hour, day of month, month and day of week, where a * means 'any'. The rest of the line is the command. For example:

```
*/20 * * * * mpg123 /path/to/sound.mp3
```

This plays the sound every 20 minutes (`*/20` is a shorthand; you could also use `0,20,40`) using the *mpg123* player. You may need to install that player through *Synaptic.* You can also restrict the hours it plays, so stop it from waking you up if the computer is switched on overnight with:

```
*/20 8-21 * * * mpg123 /path/to/sound.mp3
```



AutoKey
File Edit Tools Settings Help
New... Save Cut Item Copy Item Paste Item Delete Undo Redo Run Script
Name
Phrases
GPL3 Header C,C++ KDevelop
GPL3 Header Python PyCharm
Scripts
Citra
Gimp
close without warning
IntelliJ
Minecraft
test
(Default folder)/Scripts/Gimp/close without warning.py
Script Settings
Set
Clear

```
# Always close window without saving to Gimp's xcf
# (useful after an already successful PNG export)
import time

keyboard.send_keys("<ctrl>+w")  # Forward the captured <ctrl>+w
time.sleep(0.25)
# press <tab> 4 times, tabs keyboard focus to the "do not save" button
keyboard.send_key("<tab>", 4)
time.sleep(0.1)
keyboard.send_key(" ")  # Press "do not save" button
```

□ Always prompt before running this script
□ Show in notification icon menu

Abbreviations: (None configured)
Hotkey: <ctrl>+w
Window Filter: gimp-2\.10\.Gimp-2\.10

Autokey can be used to insert text or run commands from a key press.

Q SSH to SSH

I connect to a server with SSH and execute a command. I open another terminal and connect to the same server with the same username with SSH. How can I see in this new session what command is running in the first session?
Ed Morton

A The *ps* command lists running processes. By default, it only lists processes in the current session, but you can use **-e** to list all running processes. This is probably going from too little information to too much, so you can list only the processes of your current user with the **-u** or **--user** option. This lists each process name, but you can get more detail with **-f**, which also shows the arguments to each command. So, to give a detailed list of all processes running under the current user, we end up with:

```
$ ps -ef --user $USER
```

If you only want to see what your user is running, *ps* is the tool for the job. If you want to be able to interact with the processes, you need to prepare for this before running them. By using a terminal multiplexer, such as *screen* or *tmux,* before running a command in an SSH session, you can resume the session and interact with it from another SSH session.

Simply run *tmux* (you may need to install it first) after SSHing in, then run your

»

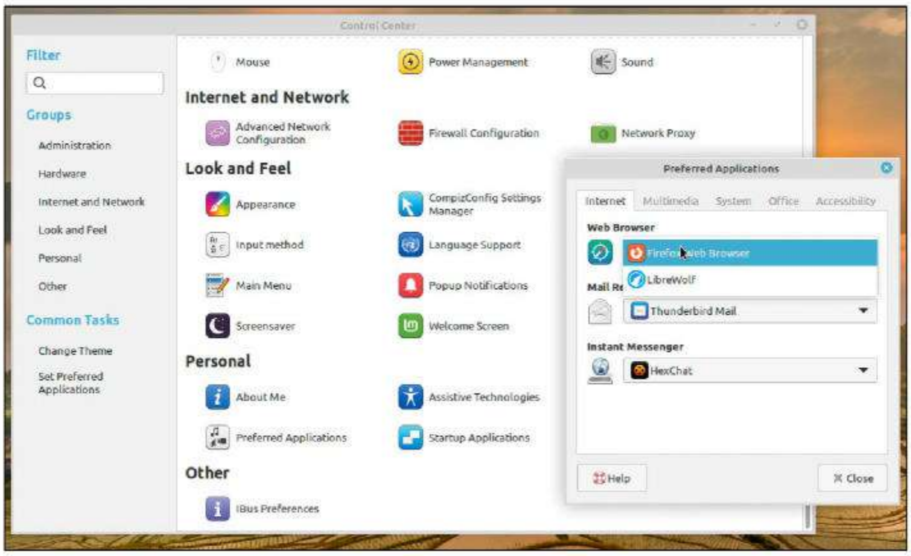
commands. The next time you SSH in, run the following command and it connects to the running *tmux* session as if you had never been away:

```
$ tmux attach
```

If, like us, you forget to start *tmux* before starting a long-running process, you can use **~/.bash_profile** to start a session automatically whenever you SSH in. Add this to **~/.bash_profile** – create it if it does not exist:

```
if [[ -n "${SSH_TTY}" ]] && [[ "${TERM}" != "tmux-256color" ]]; then
   tmux attach || tmux && exit
   fi
```

This checks if you are in an SSH session. If so, it tries first to attach to an existing *tmux* session; if that fails, it starts a new one.



Mate no longer has an option to add custom programs for filetypes, but this can be done by editing .desktop files.

Custom opening

Q **I use Mint 21.3 Mate. In Control Panel under Preferred Applications I want to change my web browser from *Firefox* to *LibreWolf,* but it's not listed in the box. How do I get *LibreWolf* listed so I can change it to my preferred browser?**
Ben Gray

A There used to be an option to add a custom handler for filetypes, but this is no longer possible. Now desktop files on the system are scanned to see which applications can handle web pages. This means you can add apps to the list by editing or replacing their desktop files.

The first step is to find the existing desktop file for *LibreWolf* with:

```
$ locate -i librewolf | grep .desktop$
```

This searches the locate database for any files containing librewolf in their name, then filters only those matches that end in .desktop. Copy the file to **~/.local/share/applications/** – files in **.local** take precedence over those in the system directories. Load the copied file into your text editor, find the MimeType line and add the types you want this program to be considered for, separated by semi-colons. You probably want something like:

```
text/html;text/xml;application/xhtml+xml;x-scheme-handler/http;x-scheme-handler/https
```

Or you could cut and paste the relevant items from the desktop file for *Firefox*. If there is no MimeType line in the desktop file, you can simply copy the whole line from *Firefox*.

Once you have done this, you may need to log out of the desktop and back in to make the change visible to the system.

This method is not limited to web browsers; you can link any program or script on your system to any list of filetypes to customise how files are opened in the file manager.

Freeing logical space

Q **How do I unallocate space from my home partition? I am currently using CentOS 7 in a 500GB HDD (50GB as root, 8GB as swap and 407GB as home). I am planning to install another Linux OS to make it dual boot. Since my home partition is 407GB, I would like to unallocate around 250GB and use it for the other OS. Is it possible? If yes, how?**

By the way, CentOS was installed as LVM. When I run *lsblk* under Type column, it is showing LVM.
Christopher Ford

A The answer to this depends on the filesystem used for **/home**. LVM was designed to make resizing storage volumes easy, provided the filesystem supports it. Unfortunately, the default filesystem used by CentOS is XFS, which does not have an option to reduce the size of a filesystem. If you use ext4 for **/home**, the process is simple, although an ext4 filesystem has to be unmounted before it can be shrunk. As it is **/home**, this is not a problem, as long as you don't open a desktop and log in to the console as root. First use *findmnt* to determine the volume group and logical volume for **/home**:

```
$ findmnt /home
```

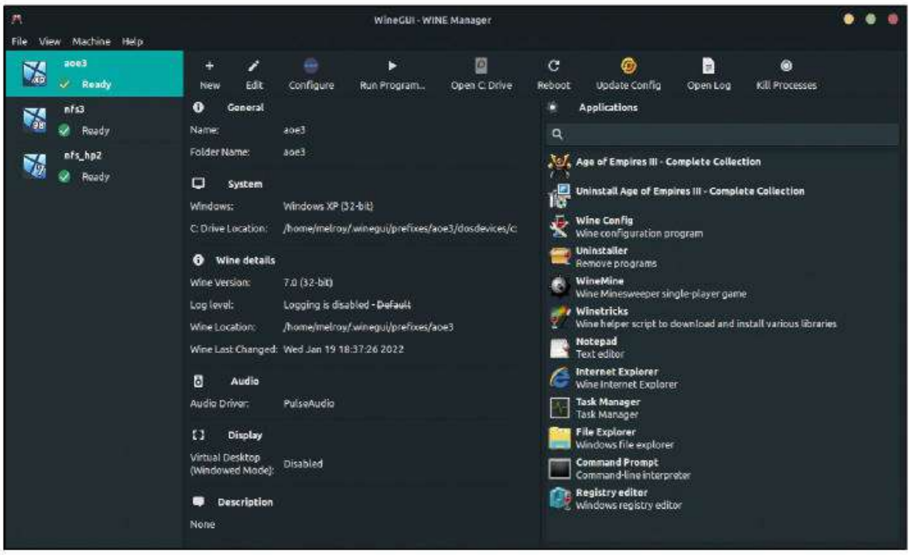
Under Source, you will see **/dev/mapper/VG-LV**, where VG and LV are the group and volume. Now unmount **/home** and resize the filesystem:

» A QUICK REFERENCE TO... FZF

Text tools such as *grep* are great at searching for and extracting text from a file or pipe, but only if you know what you are looking for. Wouldn't it be great to have a tool that let you browse the contents of a file and extract the bits you want? There is: *fzf.* It is in most distros' repos ready to install in the usual way. Once installed, *fzf* can receive input from a file or from standard input, such as:

```
$ journalctl -b | fzf
```

This loads all the system journal entries since the last boot into *fzf*. At this point, it looks like just another pager. You can move around with the cursor keys or PgUp and PgDn, but notice the text input bar at the bottom. Start typing and *fzf* jumps to the first line that matches. So far, this is nothing new, but try typing more than one search term, separated by spaces, such as `foo bar`. Now *fzf* jumps to the first line that contains either foo or bar. There are many options for search criteria, explained in the man page. Ctrl+n and Ctrl+p jump to the next and previous matches. Now press Enter to see the real magic – the program exits with the selected match sent to stdout. Selecting multiple lines is also possible. So, you can use *fzf* in a pipeline even if you don't have an exact search term. You could also use it in a script to present choices to the user. As usual, the man page contains much more information.



General
Name: aoe3
Folder Name: aoe3

System
Windows: Windows XP (32-bit)
C: Drive Location: /home/melroy/.winegui/prefixes/aoe3/dosdevices/c:

Wine details
Wine Version: 7.0 (32-bit)
Log level: Logging is disabled - ~~Default~~
Wine Location: /home/melroy/.winegui/prefixes/aoe3
Wine Last Changed: Wed Jan 19 18:37:26 2022

Audio
Audio Driver: PulseAudio

Display
Virtual Desktop (Windowed Mode): Disabled

Description
None

Wine is updated frequently and distros often fall behind with their versions. WineHQ provides the latest revisions.

```
$ umount /home
$ e2fsck -f /dev/mapper/VG-LV
$ resize2fs /dev/mapper/VG-LV 140G
```

When resizing a filesystem and then a volume, it is vital that the volume does not become smaller than the filesystem, so we play safe by making the filesystem a little smaller. Once the resize has finished, we can shrink the logical volume and then grow the filesystem to fill it:

```
$ lvresize -L 150G VG/LV
$ resize2fs /dev/mapper/VG-LV
```

Note the different way in which we specify the group and volume. Running *resize2fs* without a size resizes the filesystem to fill the volume. If you really need **/home**, you can remount it before resizing the volume, but it is more reassuring to leave that until the end.

However, if you are using XFS, there is no option to shrink the filesystem, so the only option is to copy all of the data from **/home** to a network or external drive, then unmount **/home**, resize the logical volume, create a new filesystem on it and then copy everything back. It is a pain and this limitation is why we stopped using XFS.

Q Wine nouveau

I just installed 21.3 Mate and wanted to install *Wine* in order to run a couple of Windows programs. In the software manager, there are two listed: v5.03 and v6.03. Yet when I go to WineHQ, the current stable version is v9.
Owen Woods

A Distros tend to stick with the version of a package that was stable when the distro version was tested before release and only security updates are applied. WineHQ provides repositories that include the latest stable packages. These are organised by Ubuntu code names; you can see the Ubuntu version corresponding to your Mint release with:

```
$ cat /etc/os-release
```

Look for the UBUNTU_CODENAME entry; for 21.3 it's jammy. With this info you need three commands to add the repo:

```
$ sudo mkdir -pm755 /etc/apt/keyrings
$ sudo wget -O /etc/apt/keyrings/winehq-archive.key https://dl.winehq.org/wine-builds/winehq.key
$ sudo wget -NP /etc/apt/sources.list.d/ https://dl.winehq.org/wine-builds/ubuntu/dists/jammy/winehq-jammy.sources
```

Refresh the software manager and you should see the latest versions. Bear in mind that distros test their repositories thoroughly to ensure it all works; third-party repos may not be as well tested.

Q Grubbing about for a menu

I bought a Linux computer six months ago, booted it and set a password. Now I've forgotten it. I've read about going to the GRUB menu to reset. No matter what I do, I can't get to the GRUB menu; it just goes to the enter password screen when I shut it off and back on.
Ryan Dixon

A It sounds like you have GRUB set up to boot the default menu entry with no delay beforehand, so you cannot open the menu or select anything else. This is generally a bad idea, but you don't need us to tell you that. Fortunately, the fix is quite simple. You need to boot from a USB stick or DVD into a live Linux environment. Any distro install disc that provides a live environment, such as Ubuntu, will do fine – or you could use a distro designed for system recovery, such as SystemRescue (**www.system-rescue.org**).

Boot into the live distro and open a terminal. Now you need to identify which partition holds the GRUB menu with:

```
$ sudo fdisk -l /dev/sda
```

You should see a small EFI partition, maybe a small BIOS partition and a large partition marked as Linux Filesystem. Make a note of the number then mount it, where **N** is the partition number, with:

```
$ sudo mkdir -p /mnt/linux
$ sudo mount /dev/sdaN /mnt/linux
```

Now you can load the GRUB configuration file into a text editor, most distros include *nano*, like this:

```
$ nano /mnt/linux/boot/grub/grub.cfg
```

This is a complex-looking configuration, but the change you need to make is simple. Find the line reading:

```
### END /etc/grub.d/00_header ###
```

Add these two lines after it:

```
set timeout=30
set timeout_style=menu
```

Save the file (Ctrl+O in *nano*) and reboot from your hard disk. You should see the GRUB boot menu and it waits 30 seconds for you to make a selection before proceeding with the default. The changes you made are overwritten whenever you or the package manager runs *update-grub*. To make them permanent, add these changes to **/etc/default/grub**:

```
GRUB_TIMEOUT=30
GRUB_TIMEOUT_STYLE=menu
```

Incidentally, if you do something that requires you to prefix several commands with `sudo`, save a little typing by running:

```
$ sudo -i
```

This puts you in an interactive root session, so you do not need to use *sudo* for every command. When finished, press Ctrl+D to return to your normal user. LXF

GET HELP NOW!

We'd love to try to answer any questions you send to **answers@linuxformat.com**, no matter what the level. We've all been stuck before, so don't be shy. However, we're only human (although many suspect Neil is not from around here), so it's important that you include as much information as you can. If something works on one distro but not another, tell us. If you get an error message, please tell us the exact message and precisely what you did to invoke it.

If you have, or suspect, a hardware problem, let us know about the hardware. Consider installing *hardinfo* or *lshw*. These programs list the hardware on your machine, so send us their output. If you're unwilling, or unable, to install these, run the following commands in a root terminal and send us the **system.txt** file, too:

```
uname -a > system.txt
lspci >> system.txt
lspci -vv >> system.txt
```

Subscriptions: for magazine issues email **help@mymagazine.co.uk**

Mailserver

WRITE TO US

Do you have a burning Linux-related issue that you want to discuss? Write to us at *Linux Format*, Future Publishing, Quay House, The Ambury, Bath, BA1 1UA or email **letters@linuxformat.com**.

VPNs for Linux Roundup

After reading for many years how we should use a VPN for privacy on the web, Google's recent moves persuaded me to try.

Thinking that GUI apps would be more convenient, I tried PIA, but the installer fails with a permissions error that I couldn't work around (Linux Mint 21). Despite me sending full details including the error messages, the helpdesk was useless and I had to give up and get a refund. So next I tried Surfshark, but its installer gave even more errors, and again it couldn't help so I had to get a refund.

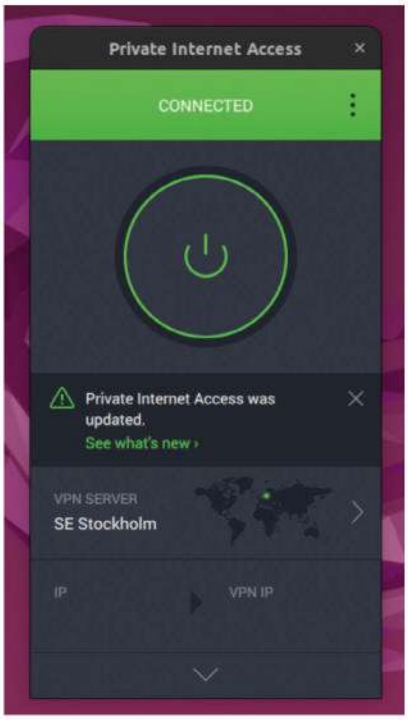
I don't know how you arrange tests on services like these, but perhaps if you gave them advance notice, they might get their Linux apps fixed before you test?

Steve

Neil says...

I've been using PIA for a few months and its installer was fine on vanilla Ubuntu. If you've had two services fail, it's possible that there is some odd configuration at your end. We are planning to run a VPN feature soon, as it continues to be a useful area for people to know about, but it's not going to highlight specific issues – companies struggle to support single versions of Windows never mind the myriad of Linux configs.

Did the PIA support mention the manual install guide? That's more likely to succeed, and if it doesn't, it might provide a more obvious reason for its failure: **https://bit.ly/lxf316pia**. The basic outline is: install *OpenVPN*, grab and unzip the configuration files, launch *OpenVPN* using the country config you want, profit. The page does offer some options if *OpenVPN* can't connect over your network/broadband connection.



Private Internet Access was updated.

See what's new ›

PIA supports Linux but that doesn't mean you won't have issues.

Repository time

Even though I have been using Linux Ubuntu 20.04 LTS now for quite a while, I still consider myself to be only an intermediate user. I seem to have problems accessing the repository in order to find apps that I can use both in my general usage and also in my ham radio hobby, such as *EchoLink*. I faithfully read my *Linux Format* magazine, cover to cover, every month and learn many things. After reading **LXF311** and deciding to install the *Dolphin* file manager, I tried to find it in a repository and was unable to do so. I actually kept looking for a way to find it and was finally able to. (I swallowed real hard, gritted my teeth and headed for the terminal, put in `sudo snap.install.dolphin` and, oh my god, it worked!) I now have *Dolphin* in my apps. I was hoping that you folks might be kind enough to do an article on how to find and use repositories.

Dr Schiro

Neil says...

Thanks to reader feedback, we are looking at more elementary topics, such as repositories and package management. This issue, we're looking at basic terminal use and package management, though not repositories – which is an entire topic in itself. But that's partly why Snaps, which you discovered yourself, and Flatpaks, which we're covering next issue, were created: to make maintaining and getting software easier. Although there are downsides to both of them, it's certainly not as bad as some segments of the internet make out.

Windows transition

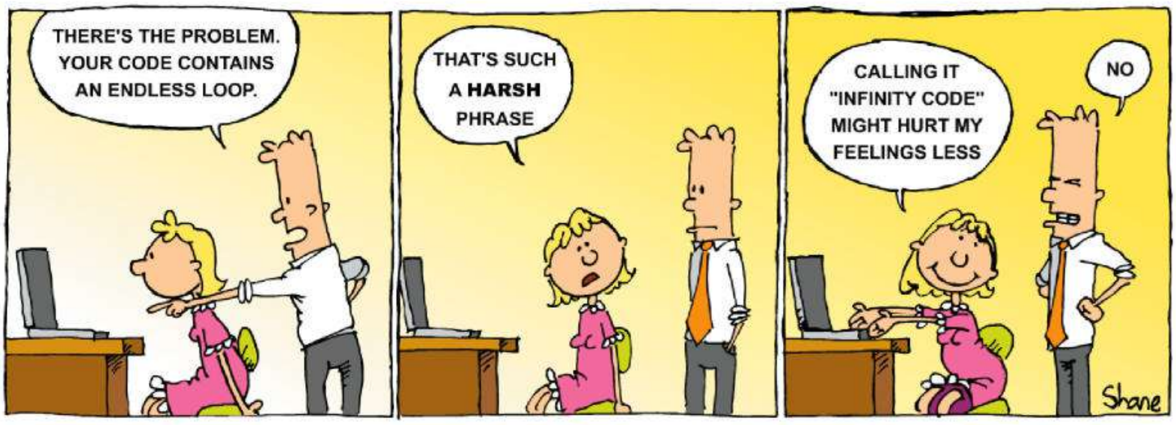
I am thinking about switching from Windows to Linux to streamline my workflow but I have some concerns. If I dual boot, will one OS affect the performance of the other? I have about 1.5TB of data I simply cannot lose but I can't afford a backup drive right now, so how big a risk is there of losing this? My version of Windows might not be genuine and I'm not sure how to back that up anyway. Any tips would be useful.

Simon Blanc

Neil says...

With modern solid-state drives, there are no performance issues at all when dual booting, just the loss of storage space. I think the real issue here is your data not being backed up. If your data is important, you need to protect that before considering upgrading or

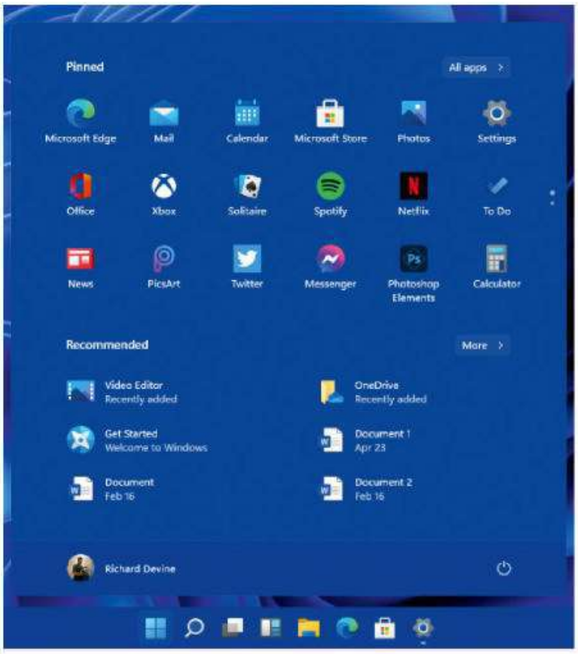
Helpdex


THERE'S THE PROBLEM. YOUR CODE CONTAINS AN ENDLESS LOOP.
THAT'S SUCH A HARSH PHRASE
CALLING IT "INFINITY CODE" MIGHT HURT MY FEELINGS LESS
NO
Shane

Flathub
Publish
Forum
About
Log in
Resources
Keep an eye on system resources
App of the Day
Citations
Manage your bibliography
Flathub is the App Store for Linux
Get the latest versions of all your favourite apps, for any Linux distribution
Set Up Flathub
Trending
Popular
New
Updated
Warp
Fast and secure file transfer
Citations
Manage your bibliography
Lapce
Lightning-fast and powerful code editor written in Rust

Flathub should be making running software easier.

changing your OS. It's annoying that drive prices have spiked this year, but 2TB drives can be picked up for around £60. I believe you can download Windows for free and use it unactivated with fairly lax restrictions, so there's not much reason to again risk your valuable data with a suspect version either, though obviously running an actual fully free operating system would be more sensible once you have backed up all your invaluable data! LXF



Plucky Redmond software outfit Microsoft uses nagware to get users to activate its paid-for desktop operating system.

LETTER OF THE MONTH

Water works

A project I'm working on – a river water level measuring system using ultrasonics – formed this idea for an article. With the range of Raspberry Pi units from the simplest/cheapest to the most complex/dearest being so wide, it has become difficult to appreciate which Pi is the cheapest/least power-hungry but still capable enough to achieve different purposes. Then there is the alternative Arduino stream of microprocessor boards.

In my case – looking to record ultrasonic distance measurements probably every minute, with 30-minute averages, powered by small battery, with or without Wi-Fi link via phone card, backup storage on SD card and possible maintenance only every month – which Pi do I use? Is a Pi even suitable – perhaps I should be looking at Arduinos?

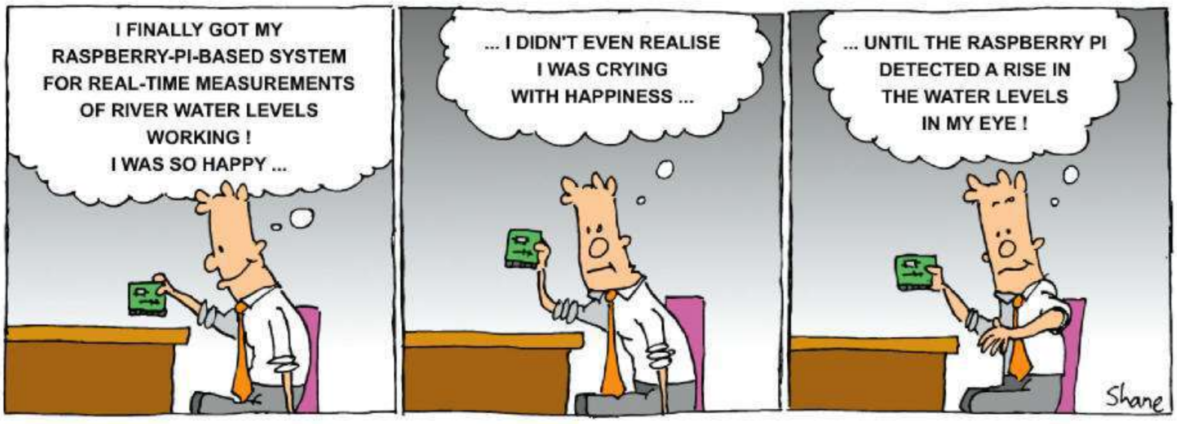
Very few online projects cover this aspect – an aspect that is crucial in my line of work where a multitude of surface sensors must be interrogated and logged in different environments, from Arctic to desert to tropical rainforest, each with different climates and visit timings (the Arctic ones had to run for several months through the polar night) that required different power supply options and logging alternatives. Admittedly, using commercial electronic logging systems from firms such as Campbell Scientific meant that you could be assured of reliability in the face of a severe climate.

Dr Colin

Neil says...

If you want a month-long battery life, I'd just go straight to the Pi Pico. The documentation (**https://bit.ly/lxf316pico**) states 1.3mA in sleep, 0.92mA in dormant and 9.8mA in bootsel mode with the USB bus active. This is at its normal 5V. Using four AAs at 2,400mAh (being generous) that would be 76 days in sleep or 10 days active. Assuming you could fall to sleep mode between readings, you might be able to eke out almost a month of battery, although using the Pico would require a little more effort to implement a solution. The problem with a 'full' Pi is that even the Pi Zero demands 100mA power consumption, so long-term battery life is a challenge. Good luck with your project and do let us know how you get on.

If you want devices running for multiple days, the Pico can help out.


I FINALLY GOT MY RASPBERRY-PI-BASED SYSTEM FOR REAL-TIME MEASUREMENTS OF RIVER WATER LEVELS WORKING ! I WAS SO HAPPY ...
... I DIDN'T EVEN REALISE I WAS CRYING WITH HAPPINESS ...
... UNTIL THE RASPBERRY PI DETECTED A RISE IN THE WATER LEVELS IN MY EYE !
Shane

shane_collinge@yahoo.com

SUBSCRIBE TO LINUX FORMAT MAGAZINE

SAVE 50%

www.magazinesdirect.com/LIN/D35T

Call **0330 333 1113** and quote **D35T**

FREE DVD! UBUNTU + FEDORA UPDATED!
LINUX FORMAT
Get into Linux today!
GET STARTED WITH LINUX! Beginner's guide
RASPBERRY Pi 2! 500% FASTER
EXCLUSIVE! Hands-on guide and in-depth review of the all-new Pi 2
Made of Wynn
GET CREATIVE TODAY!
FREE DOWNLOAD MINT 20.2
CUSTOMISE MINT 20!
67 pages of tutorials & features
GET INTO CODING!

» NEW: Improved digital back-issue access!**

Get **instant access** on your browser, or Android and Apple iOS devices back to issue 181 (March 2014) with 1,000s of tutorials, interviews, features and reviews.

CHOOSE YOUR PACKAGE!

Keep your subscription rolling!

Subscribe today and save 50%!* + All-new digital access**

NEW! Digital access to 130+ issues when you subscribe to print!**

PRINT EDITION

Only £21.09

every 6 months

6 print issues of *Linux Format* every 6 months delivered to your door for £21.09!

DIGITAL EDITION

OUTSIDE THE UK? Turn to **page 61** for more great subscriber deals!

Only £22.71

every 6 months

6 digital issues of *Linux Format* every 6 months for £22.71!

Terms and conditions: **Offer closes 25th June 2024**. Please allow up to 6 weeks for the delivery of your first subscription issue (up to 8 weeks overseas). The full subscription rate includes postage and packaging. *Savings are based on the cover price. Payment is non-refundable after the 14-day cancellation period. **Access to the digital library will end with your subscription on print. For a digital subscription, once terminated you only keep the issues you have paid for up to that point. For full terms and conditions, visit www.magazinesdirect.com/terms. For enquiries and overseas rates please call: +44 (0) 330 333 1113. Lines are open Monday-Friday 8:30am-7pm, Saturday 10am-3pm UK time (excluding bank holidays) or email: help@magazinesdirect.com. Calls to 0330 numbers will be charged at no more than a national landline call, and may be included in your phone provider's call bundle.

TAKE YOUR LINUX KNOWLEDGE FURTHER

Enhance your knowledge with in-depth projects and guides

HACKER'S MANUAL 2024
ON SALE NOW
FULL OF EXPERT TIPS & ADVICE
ADVANCE YOUR LINUX SKILLS
• THE KERNEL • NETWORKS
• SERVERS • HARDWARE • SECURITY
148 PAGES OF TUTORIALS
ENHANCE YOUR KNOWLEDGE WITH IN-DEPTH PROJECTS AND GUIDES
Hacker's Manual 2024 Sixteenth Edition
ISBN 978-1-80521-409-0
FUTURE

FUTURE

Ordering is easy. Go online at:

magazinesdirect.com

Or get it from selected supermarkets & newsagents

SanDisk Pro-Blade

Flashing back to Zip drives, **Ganesh TS** is perplexed.

SPECS

Size: 4TB (1TB, 2TB options)
Port: USB 3.2 Gen 2x2 Type-C, PCIe 3.0 x4
Bridge: ASMedia ASM 2364
Power: Bus
Cable: 50cm Type-C
Size: 109.8x28 x7.5mm 45g (Mag), 130.4x 71.5x16mm 165g (Transport)

The Transport caddy with the SSD Mag sticking out the end – it's a rugged little thing.

Western Digital unveiled the mouth-filling SanDisk Professional Pro-Blade modular SSD ecosystem in mid-2022 to serve the needs of the professional market. The Pro-Blade family comprises of three product lines: compact and sturdy NVMe drives (Pro-Blade SSD Mags), a USB 3.2 Gen 2x2 (20Gb/s) enclosure for the mags (Pro-Blade Transport), and a four-bay Thunderbolt enclosure (Pro-Blade Station) directly compatible with the mags. The mags are essentially M.2 2280 NVMe SSDs with a proprietary connector that directly exposes the internal PCIe Gen3 x4 interface, albeit with support for a higher number of mating cycles compared to the native M.2 2280 internal SSD one.

We're taking a detailed look at the internals of the Pro-Blade SSD Mag and the Pro-Blade Transport enclosure, followed by a discussion of the results of putting the 2TB and 4TB combinations through a rigorous direct-attached storage performance evaluation routine.

Are we going mag?

The Pro-Blade SSD Mags are available in 1TB, 2TB and 4TB capacities. The mags have a gumstick form factor. They are essentially a thermal cladding for an M.2 2280 NVMe SSD with a customised PCIe Gen3 x4 connector and provide a snug fit.

The enclosure has a stylish and solid feel to it, with the ridges on the side giving users additional traction. Western Digital utilises an ASMedia ASM2364 bridge chip under the thermal pad in the Transport enclosure's board. The SSD Mag incorporates the Western Digital WD_Black SN750E M.2 2280 Gen3 x4 NVMe SSD with plenty of thermal paste ensuring good contact with the metal sides. This SSD uses an in-house SanDisk controller along with 3D TLC NAND (BiCS 4 96L 3D TLC) and DRAM for the flash translation layer.

Flash-based storage devices tend to slow down in unpredictable ways when subjected to a large number of small-sized random writes, though such workloads are uncommon for direct-attached storage devices, where workloads are largely sequential in nature.

The 4TB version can write around 60GB at 1.8GB/s-plus before moving to a direct-to-TLC write speed of 1.6GB/s. On the other hand, the 2TB version can write around 28GB at 1.8GB/s-plus before settling down to 1.5GB/s for the rest of the workload. Temperatures end up at 56°C and 52°C for the two variants. The performance profile of the 4TB variant matches that of the SanDisk Extreme Pro v2 of the same capacity. The Crucial X10 Pro 4TB is able to maintain 1.6GB/s throughout, but it doesn't have the initial 1.8GB/s-plus burst. Other portable SSDs (PSSDs) suffer from significant performance loss in the duration of the test. Overall, this performance profile is ideal for the sustained heavy write workloads seen in the content capture industry.

The 4TB version has a peak power consumption of 9.58W, while the 2TB one peaks at 8.26W. Fortunately, there is an idling mode for both, where the power consumption drops to 0.34W. Bridge-based solutions incorporating SSDs with DRAM are expected to be power hungry, and the Pro-Blade Transport configurations are no exception. Fortunately, the devices provide excellent performance consistency to go along with this power profile.

Based on the evaluation of different portable SSDs and the current pricing, the performance of the Pro-Blade SSD Mag/Transport is very compelling. The thermal design is top notch, and the value proposition is excellent for professional users. Casual users and even prosumers may probably be better served by cheaper PSSDs that are more power efficient. LXF

VERDICT

DEVELOPER: Western Digital
WEB: www.westerndigital.com
PRICE: £577 4TB (£216 1TB)

FEATURES	8/10	EASE OF USE	9/10
PERFORMANCE	9/10	VALUE	7/10

Fitting standard M.2 drives into non-standard housing seems counterproductive, but performance and build are excellent.

» **Rating 8/10**

Archcraft 2024.04.06

Daring **Nate Drake** takes a trip in the Archcraft and finds it flies at the speed of light.

IN BRIEF

Archcraft's lack of traditional desktop doesn't stop its interface being a joy to use. The integrated welcome guide and array of customisation features also make it ideal for those new to Arch-based distros.

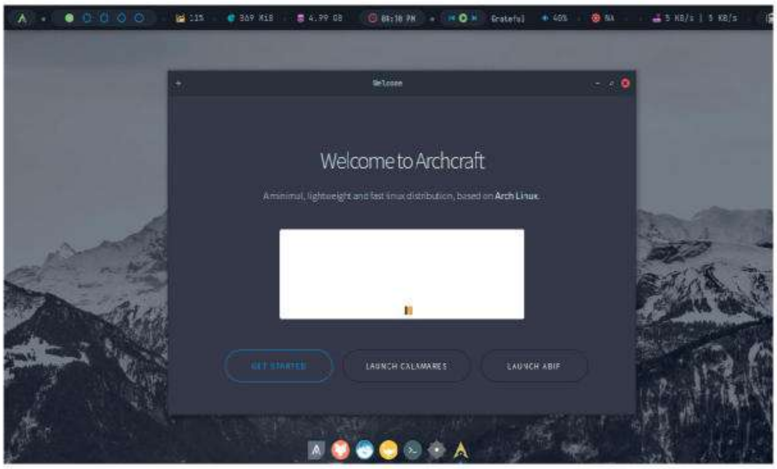
SPECS

CPU: 1GHz
Mem: 512MB
HDD: 20GB
Builds: x86_64, ARMv7, ARMv8

Arch Linux is well known for being a lightweight distro, but since 2021, developer Aditya Shakya has taken it a step further with his own derivative creation, Archcraft.

This distro keeps itself speedy by eschewing the traditional desktop environment in favour of window managers – *Openbox* for stacking and *bspwm* for tiling.

Archcraft also supports a number of Wayland compositors such as *Sway* and *Wayfire*. These can be installed manually, but the main website also offers a premium version of the OS, Archcraft Prime. This comes with the premium version of the *Openbox*, *i3* and *bspwm* window managers, as well as the *Sway, Wayfire, River, Hyprland* and *Newm* Wayland compositors preinstalled. A minimum 'donation' of $55.55 is required. We're focusing on the community edition, but you can read more and download the Prime version at **https://ko-fi.com/s/c925a2a8c1**.


Welcome
Welcome to Archcraft
A minimal, lightweight and fast linux distribution, based on Arch Linux.
GET STARTED
LAUNCH CALAMARES
LAUNCH ABIF

You can use Archcraft's welcome guide to view documentation, such as the official wiki, or launch the Calamares or ABIF installers.

Like Arch, Archcraft OS follows a rolling release model. The latest version (2024.04.06) has a number of updates for both packages and window managers.

We were keen to try Archcraft in a live environment, so proceeded to download the modestly sized Live ISO. Website visitors have the choice of direct download via Sourceforge, Google Drive or Mega. We were surprised not to see a BitTorrent option but persevered.

On first launch, we noted that Archcraft uses the GRUB2 bootloader. This has received some attention in the latest release, which now ensures that GRUB is installed on every update.

We were also delighted to see that the desktop loaded within seconds, launching a helpful welcome guide. This is where we learned that Archcraft uses the intuitive *Calamares* installer for setup. According to the release notes, it has been updated to the latest version.

Alternatively, you can deploy a secondary installer, *ABIF,* though as this is command-lined based, the official wiki advises it's not for beginners.

Links to the wiki are available by choosing Get Started in the welcome guide. You can also access tutorials and a gallery of images of the Archcraft UI.

In keeping with Archcraft's minimalist philosophy, the lower dock contains only six icons. The first of these is for the excellent OpenGL terminal emulator *Alacritty.* Web browsing is provided care of *Firefox,* which has been updated to version 124.0.2.

The third icon launches *Thunar,* Archcraft's file manager of choice. The latest release of Archcraft has made some minor tweaks, allowing permanent setting of wallpapers, as well as the ability to open as terminal or as root in supported sessions.

The fourth icon is assigned to the Xfce terminal, which seems a little redundant given the presence of *Alacritty,* while the final icon is given over to *Settings.*

We decide to launch Settings to see if we could do anything about the austere grey theme for the default desktop. Upon choosing Appearance, we discovered there are indeed a huge number of alternative Styles available, including the far greener EverForest. There's also a staggering 65 alternative wallpapers available.

We rounded off our test drive of Archcraft by using *Alacritty* to fire up the *yay* package manager, to install *Mozilla Thunderbird* in less than 10 seconds by entering `yay -S thunderbird`.

If you have time to kill, take a moment to examine the top panel. Here you can play one of the preinstalled tunes via the mini media player. You can also view vital information such as download and upload speeds, and stats on system resources. LXF

VERDICT

DEVELOPER: Aditya Shakya
WEB: https://archcraft.io
LICENCE: GPL 3.0 (free version)

FEATURES	8/10	EASE OF USE	8/10
PERFORMANCE	9/10	DOCUMENTATION	8/10

Offers a blazingly fast interface via its minimalist desktop and handful of preinstalled apps, while being highly customisable.

» **Rating 8/10**

Fedora 40

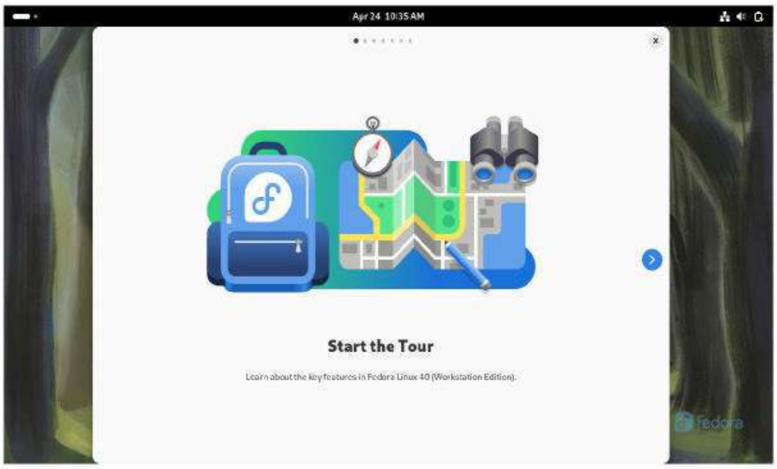
Nate Drake doffs his hat to the very latest version of Fedora as he gives the Workstation edition a twirl.

IN BRIEF

Fedora has a reputation for innovation and version 40 doesn't disappoint, with new tools, upgraded packages and shiny new desktop environments. Bundled apps aren't copious – but you can easily install more.

SPECS

CPU: 2GHz dual-core
Mem: 2GB (4GB recommended)
HDD: 15GB (20GB recommended)
Builds: x86-64, armhf, aarch64



Apr 24 10:35 AM

Start the Tour

Learn about the key features in Fedora Linux 40 (Workstation Edition).

Fedora 40 ships with Gnome 46 by default. There's also a KDE edition, which uses the Plasma 6 desktop environment.

Fedora is developed by the community-based Fedora project. It's the upstream source for both CentOS Stream and RHEL (Red Hat Enterprise Linux).

There are two major OS releases every year, targeted for the fourth Tuesday in April and October. Releases are supported for just 13 months. This review focuses on Fedora 40, which at the time of writing is still in beta, so we encourage you to download and test features for yourself.

This is easier said than done, as visitors to the main website will find there are five editions of flagship Fedora alone: Workstation, Server, CoreOS, IoT and Cloud. It is also available in a variety of spins, such as the immutable Atomic Desktops, and those with alternative desktop environments like KDE, Cinnamon, Budgie and Xfce.

We went mainstream by downloading the 2GB Workstation ISO, which uses Fedora's default Gnome desktop environment. The latest version of the OS uses Gnome 46, so includes enhancements like the new global search feature, as well as an overhauled file manager and better support for online accounts.

Users of Fedora 40's KDE spin will also discover the OS now ships with the Plasma 6 desktop environment and uses Wayland. The KDE team, however, promises that X11-native apps will still run under Fedora 40.

After booting Fedora 40 into the live environment, we launched the virtual tour of its latest features via the dedicated app. Here we learned that you can simply tap the Super key to display an overview of windows and apps. The app also provides guidance on how to use trackpad gestures to navigate workspaces.

The Fedora Project website also has an extensive documentation section (**https://docs.fedoraproject.org/en-US/docs/**), containing release notes, installation instructions and a series of tutorials to get started easily. We've given this special mention because, with the exception of Red Hat, this is the most extensive online documentation (*ArchWiki wants a word–Ed*) we've seen for a Linux distro.

The Fedora Project also has a comprehensive wiki, which is where we were able to view the Change Set for Fedora 40. This is where we learned that GoLang packages have been updated to version 1.22, as well as the fact that the integrated network manager can now automatically detect IPv4 address conflicts. While we're talking connectivity, Fedora also now makes use of stable-SSID as the default mode for assigning individual, stable MAC addresses to Wi-Fi connections in *NetworkManager,* enhancing both stability and user privacy.

Other upgrades include bundling java-21-openjdk, as well as Boost 1.83. The AMD ROCm accelerator framework has been updated to version 6. Fedora also has an updated toolchain, incorporating gcc 14.0, binutils 2.41, glibc 2.39 and gdb 14.1.

Image Builder is also now being used to build the Fedora Workstation Live ISO. The Fedora Project hopes this will reduce issues in the image building pipeline, due to Image Builder's more comprehensive testing relative to alternatives like lorax-composer.

In terms of apps, Fedora 40 Workstation comes with the project's own build of *Firefox* (v124.0.1). The OS also bundles the latest *LibreOffice* (v24.2) apps.

Other than some of Gnome's core apps, such as the *Rhythmbox* music player and *Boxes,* that's the extent of Fedora's basic configuration. Fortunately, these default apps include *Gnome Software,* which we used to download and install the Flatpak version of *Mozilla Thunderbird* email client via Fedora's official repos. LXF

VERDICT

DEVELOPER: Fedora Project
WEB: https://fedoraproject.org
LICENCE: Mainly GPL

FEATURES	8/10	EASE OF USE	9/10
PERFORMANCE	9/10	DOCUMENTATION	10/10

Fedora 40 is all that makes this OS great: simple setup and clockwork precision in selecting and upgrading apps.

» **Rating 9/10**

Austrumi 4.9.3

Nate Drake discovers how a distro can be both fast and clunky at the same time when booting Austrumi.

IN BRIEF

Austrumi lives up to its claims by booting directly into RAM and offers a surprising number of web and productivity apps. It's let down by a clunky interface and little documentation.

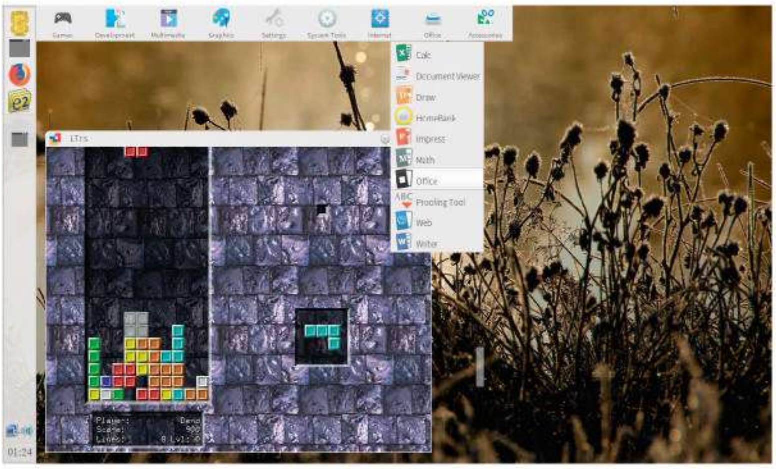
SPECS

CPU: 1GHz 64-bit
Mem: 1GB (2GB recommended)
HDD: 500MB
Builds: x86_64

Most Linux users are aware of Slackware; not only is it currently the oldest actively maintained Linux distribution, but it's also designed for simplicity and interoperability.

Slackware's lack of graphical installer and pre-installed desktop environment only make it suitable for advanced users, hence the number of distros that are based on the operating system.

Austrumi has sought to square this circle since 2009, when the first stable version was released. This Latvia-based distro's main selling point is that it's capable of booting from a CD/ISO, then loading the entire OS and associated applications into RAM, enabling users to remove the boot medium.



Austrumi has a surprising number of bundled programs for a minimalist distro, including LibreOffice 7.6 and an assortment of games.

In theory, this has many advantages for privacy, given that once the device reboots, any data stored in virtual memory soon dissipates.

We decided to download the minimal ISO from Austrumi's main site, which is where we hit our first hurdle. The Download link on the website seems to point to a non-existent FTP server. Having tried this out in both *Firefox* and *FileZilla*, we found Distrowatch had a link to the main folder (**https://ftp.linux.edu.lv/austrumi/**), which we used to download the latest version of Austrumi (4.9.3).

Upon first boot, users are presented with four options. The first is to boot the OS into RAM then eject the boot medium. The second option is to retain the boot medium, which is recommended for machines with 1GB of RAM or less. The third option involves logging in as the root user (also ejecting the boot medium), and the fourth is for text mode.

Upon first log in, we hit another snag in that while Austrumi supports 15 languages, the default is the native tongue of the developers: Latvian. We enlisted the help of a Latvian friend to discover you can fix this, first by hovering over the gold icon at the top-left, then choosing Istatiejumi (Settings) from the top panel that appears. From here you can choose Volūdys (a word we assume to be 'languages') to switch languages.

Doing so restarts the X server, which prompts the user to log in again. This is one of the occasions where the Austrumi forum could have come to the rescue, if it weren't for the fact that the link was dead. (From reading other online reviews, this seems to have been the case for some time.)

Fortunately, we were able to discover the user password was the same as the root one ('austrumi') by restarting and cycling through the boot options, where it was listed.

Once Austrumi became intelligible, we found that it does live up to its reputation for being lightweight, as we right-clicked the desktop to choose from one of nine alternative backgrounds.

From examining the Settings in English, we found that Austrumi can actually be installed to a hard drive, though no details are given on how large the install footprint is. In the specs *(see left)*, we've assumed it's roughly equivalent to the size of the ISO (430MB)

For such a small distro, there is a surprising number of bundled applications. These include no fewer than 14 games, including *Sudoku* and *LTris*. Austrumi also bundles the hex editor *GHex* and can emulate machines via QEMU through *Virtual PC*. Web browsing is managed by *Firefox* (v124.0.2), which upon launch displays a local help page (again written in Latvian).

Users can also visit the Office section of the top panel to launch any of the *LibreOffice 7.6* apps such as *Writer* as well as the integrated *Document Viewer*. LXF

VERDICT

DEVELOPER: Andrejs Meinerts
WEB: http://cyti.latgola.lv/ruuni
LICENCE: Various

FEATURES	7/10	EASE OF USE	3/10
PERFORMANCE	7/10	DOCUMENTATION	1/10

Austrumi may be fast but it's plagued by a lacklustre and buggy interface. Consider alternatives such as Slax.

» **Rating 5/10**

Calculate Linux

Nate Drake crunches the numbers with Calculate Linux and finds it earns top marks for its excellent performance and features.

IN BRIEF

CLD makes the Gentoo experience much easier, offering desktop flavours and an intuitive installer. Default apps are thoughtfully selected but you need to be comfortable with the CLI to install more.

SPECS

CPU: 1.3GHz 64-bit
Mem: 1GB
HDD: 8GB
Builds: x86_64

Calculate Linux has been around since 2007. The project website states it's also backward compatible with its parent OS, which we assume to mean Calculate has full binary compatibility with Gentoo.

The website is, in fact, home to three main editions of Calculate. These include Calculate Directory Server (CDS), which can act as a domain controller. It is also capable of configuring Samba, mail, XMPP and proxy services. Calculate Linux Scratch (CLS) is intended for advanced Linux users. It consists of a live CD with a build framework for creating a custom distribution.

Calculate Linux Desktop (CLD) is the workstation/client edition and is the focus of this review. There's no single official desktop environment; users are offered a choice of flavours with Cinnamon, KDE, LXQt, Mate, Xfce. The system specs *(right)* are for the Cinnamon edition, which we used. The LXQt, Mate and Xfce editions require only 512MB of RAM.

After downloading the 3GB ISO, we booted the OS, only to hit a snag as we were prompted for the guest password. Luckily this was simply the word 'guest', which we entered to load the Cinnamon desktop.

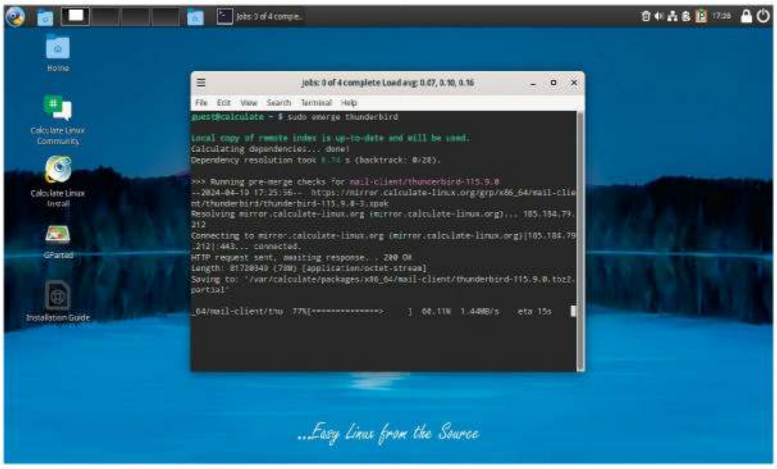
We were particularly intrigued by this latest version of CLD (v20240401), given the announcement in the user forums in early April that the OS would be going release-free. In other words, Calculate's nightly builds will officially become main releases, following a similar model to Arch Linux and Gentoo itself.

According to lead developer Alexander Tratsevskiy, this approach has a number of advantages, including that any time spent preparing new fixed releases can be better devoted to improving the OS, not to mention bleeding-edge updates. As version 20240401 was seemingly released before this announcement, we concluded it had just escaped this proclamation.

The Calculate Forum also includes links to the official wiki, which provides excellent introductory setup and configuration tutorials. It was here that we also learned that CLD supports installation to both USB drives and hard disks using the ext4, ext3, ext2, ReiserFS, Btrfs, XFS or JFS filesystems.

If you need further hand-holding, on first boot the desktop contains a shortcut to the *Polari* client, which automatically launches Calculate's dedicated IRC channel. Sadly, this didn't load for us, though this may be because we were using CLD's live environment.

Polari's failure doesn't detract, however, from the comprehensive set of out-of-the-box apps. Some of these are accessed via the bottom dock, including CLD's browser of choice *Chromium.*



```
guest@calculate ~ $ sudo emerge thunderbird

Local copy of remote index is up-to-date and will be used.
Calculating dependencies... done!
Dependency resolution took 0.74 s (backtrack: 0/20).

>>> Running pre-merge checks for mail-client/thunderbird-115.9.0
--2024-04-19 17:25:56--  https://mirror.calculate-linux.org/grp/x86_64/mail-clie
nt/thunderbird/thunderbird-115.9.0-3.xpak
Resolving mirror.calculate-linux.org (mirror.calculate-linux.org)... 185.184.79.
212
Connecting to mirror.calculate-linux.org (mirror.calculate-linux.org)|185.184.79
.212|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 81720349 (78M) [application/octet-stream]
Saving to: '/var/calculate/packages/x86_64/mail-client/thunderbird-115.9.0.tbz2.
partial'

_64/mail-client/thu  77%[===============>     ]  60.11M  1.44MB/s    eta 15s
```

Just as with Gentoo, CLD package management is handled by Portage. Search for and install packages via the command line using emerge.

From here, users can also launch the *Evolution* email client, as well as version 7.6.4 of the *LibreOffice Writer* and *Sheets* apps (the other *LibreOffice* programs are accessible via the main menu). There are also shortcuts to the default *Eye of Gnome* image viewer, as well as the ever-awesome *GIMP* image editor.

We couldn't find *Software Center* but the wiki says that package management is handled by *Portage*. As in Gentoo, you can search for and install software via the terminal using the *emerge* command. We did this to install *Mozilla Thunderbird* in moments, although CLD seemed unable to find *Kuroo*, a graphical front-end for *Portage,* even though it resides in the main tree.

We were, however, very impressed by Calculate's own installer. Not only does it offer the aforementioned filesystems but it supports using separate ones for the **root** and **data** partitions. We couldn't see any options for full disk encryption, but the wiki states that **home** folder encryption is supported. LXF

VERDICT

DEVELOPER: Calculate Linux team
WEB: www.calculate-linux.org
LICENCE: Mostly GPL

FEATURES	9/10	EASE OF USE	8/10
PERFORMANCE	8/10	DOCUMENTATION	7/10

Offers a variety of desktops and easy setup. Our only major complaint is there's no graphical package manager.

» **Rating 8/10**

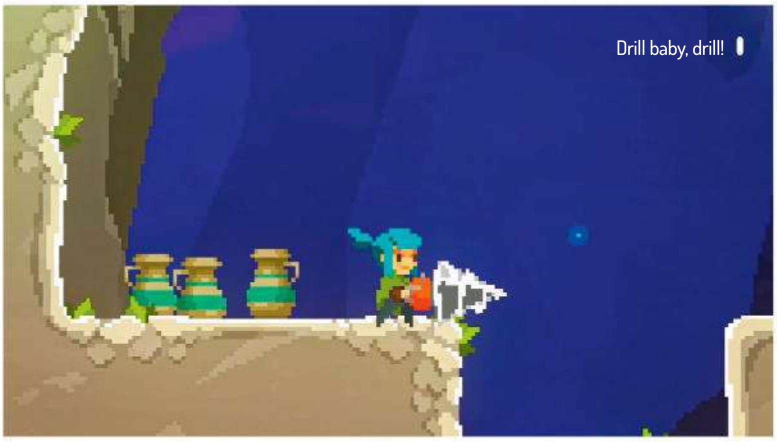
Pepper Grinder

Abbie Stone finds management is thrilled to have drill-equipped staff!

SPECS

Minimum
OS: Ubuntu 22.04
CPU: Intel Core i5 4570, AMD A10-5800K APU
Mem: 8GB
GPU: GeForce GTX 650, Radeon HD 7750, Iris Pro Graphics 580. OpenGL 2.1+, 1GB VRAM
HDD: 300MB

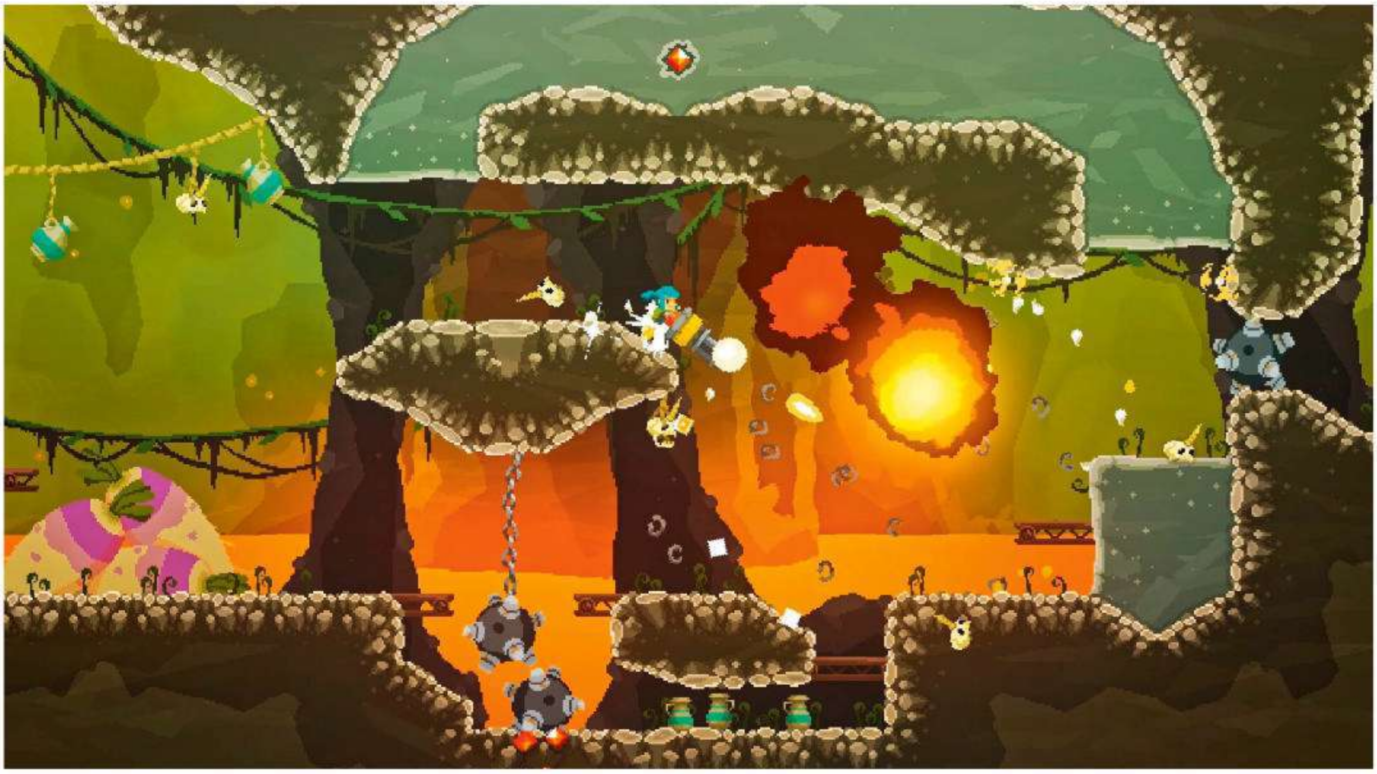
Poor Pepper. She's the star of Devolver Digital's new platformer, but she would be laughed off the stage at the Annual Platforming Protagonist Awards. Not that she'd have much luck leaving that stage unassisted. Pepper's hobbled with a pathetic jump, a mediocre little hop that can barely get her a few feet off the ground. So, it's a stroke of good fortune that she's rammed her arm into a drill attachment that can send her swimming through dirt, then use her momentum to burst out and fly through the air.


Drill baby, drill!

Well, maybe less 'good fortune', more 'mixed blessing' and perhaps even 'get this damn monkey's paw off me'. It turns out that having a drill for an appendage can be quite cumbersome. Pepper drills through dirt a little faster than you'd like, demanding quick reflexes to move in something close enough to the direction you wanted to go, or at least a route that ideally doesn't end in certain death. This is all completely by design, of course, and gradually mastering such a joyously unwieldy method of navigation makes for a terrific little platformer.

Cumbersome as it can be, the drill is a delight to use, with excellent rumble feedback. This is a must-play with a controller, unless you own a vibrating keyboard. (Why do you own a vibrating keyboard? Actually, we'd rather not know.) Swimming through dirt feels great, as does introducing enemies to the sharp spinny end of your new toy. Comparisons to Game Boy Advance cult classic *Drill Dozer* are obvious, but it actually plays more like that same console's *Donkey Kong: King of Swing*. That game was all about carefully timing swings to reach higher ground. *Pepper Grinder* plays a bit like this, if you were playing *King of Swing* with a dodgy emulator that quadrupled the game's speed.

To exit a swing, you need to trigger the drill, sending Pepper skyrocketing. Levels soon become an acrobat's nightmare of perfectly executed chains of swinging, drilling and hurtling yourself around. There's even a speed boost that's tragically mandatory to execute some


When is a drill not a drill? When it's a gun.

Levels are lavish fun with your equipped drill.

of the more distant jumps. Pepper has four slivers of health that can disappear in the blink of an eye and a tiny window of invulnerability after taking damage. Thank Christmas there's no lives system and mostly generous checkpointing, so this tricky platformer usually lands on the right side of challenging/frustrating. When you pull off a long, unbroken run of platforming, it gives you a similar sense of triumph to racing through a 2D *Sonic* level without dropping a ring, except success here feels much more skill-based.

Pepper Grinder breaks up this demanding platforming by offering gloriously silly extensions to your drill arm. You occasionally find a gun with infinite bullets that turns the game into *Gunstar Heroes* with the cheats on. Even those sections play like *Elden Ring* compared to the giant mech suit that lets you smash through the scenery. At first these sections feel like pay-offs for surviving, a few seconds of cathartic, brain-off destruction. But later levels weave the gadgets into the challenge, taking advantage of how vulnerable Pepper is without them by contriving circumstances that separate you. A few minutes of panicky, tough platforming are all the better for being punctuated with another go inside the mega death mech.

No driller filler

There's a grassy area, lava area, ice area… Hey, wake up! True, these are among the most tired locations in platforming history, but there's enough visual variety and quirks to keep things interesting. Like a level full of friendly giants assisting you, where you at one point have to drill through one of their heads (this is a lot less gross than it sounds). The platformer standard water level is livened up by having goons on fragile boats that you sink one by one with spinning drill leap-attacks. It's constantly throwing novelties like this at you, never feeling like it's cynically padding out the run time with filler.

There are some minor issues. It crashed a few times and one boss froze in place on two attempts, presumably intimidated by how well we were playing. Nothing a few patches can't fix, of course. One missing UI element that teaches you how to swing has already been patched in, so these fixes might already be in the game by the time you read this. A few more checkpoints wouldn't go amiss either, as some of the late-game levels boot you back far enough to have us fantasising about introducing the drill to the developers' backsides, but now we're just being childish. And creepy. Er, sorry.

Replayed levels have time trial trophies, a good way to test your navigation skills under pressure, though we suspect watching a speedrunner glide through this game will depress us enough to put our fingers in a retirement home. There are five collectable coins to be discovered in each level and we like that if you find one and then die later on, the game doesn't pointlessly punish you by making you collect it again. Among other rewards, these coins unlock a few bonus levels, adding a little longevity. There's a sticker album you can gradually fill out if you want to make your own little *Pepper Grinder* dioramas, and you can buy different haircuts and capes, but these are throwaway extras.

Which is honestly fine! It's not a long game – it took us five hours to see the credits – but it's generous with ideas right up to those credits. *Pepper Grinder* is a short, sweet treat, a good idea executed with great energy. If you're looking for a 2D platformer built to last, get *Rayman Legends.* If you're up for a faster, leaner platformer, crammed with memorable moments (Pumpkin carving! An homage to *King Kong!* An absolute pig of a final boss!), Pepper's thrilling drilling adventure is well worth a spin. LXF

Do you fly or swim through dirt?

VERDICT

DEVELOPER: Ahr Ech, MP2 Games
WEB: www.peppergrindergame.com
PRICE: £12.79

GAMEPLAY	9/10	LONGEVITY	5/10
GRAPHICS	7/10	VALUE	8/10

Five hours of fun, frantic platforming that's unwieldy and all the better for it.

» **Rating 8/10**

Roundup

Visual Studio Code » LibreOffice Writer » LyX » Google Docs » FocusWriter

Michael Reed could write a book about his experiences with Linux and the open source world.

Editors for authors

Whether you're working on a fiction or non-fiction book, **Michael Reed** believes it's vital to have an editor that works the way you want.

HOW WE TESTED...

Fortunately, the author happened to be working on a fiction book project while writing this article. This made it possible to shift the main body text between the different editors and put in some serious hours of text creation to get a true picture of what each editor was like to really use on a book-length project. While working, the project was moved between a desktop and a less powerful laptop to make sure that the systems could work with a smaller screen and less powerful CPU.

In the case of going from Markdown format to ODF format, we used the *Pandoc* tool. This enabled us to import into both *LibreOffice* and *FocusWriter.*

The *Visual Studio Code* solution required the most configuration work because it was a DIY setup. There was a bit of detective work and experimentation needed to find the best extensions for word count, Markdown support, grammar and spellchecking, and sharing profiles between computers.

This issue, we're looking at document processors for authors that can be used to write a long, text-based project such as a book. Some you'll know but other might seem a little leftfield such as *Visual Studio Code*, better known as a code editor, but with the right extensions, it can be transformed into a perfectly good writing environment for a book. It's a bit of a DIY solution, but it worked well in our tests.

The *LibreOffice Writer* module is a traditional word processor with tons of features and integration with the rest of the office suite. On modern hardware, it's lightweight and fast, and it can turn an older laptop into a powerful writing workstation.

LyX sits on top of the LaTeX markup language. It can reflect things such as font weights on screen, even though it's not quite a WYSIWYG editor. You may never see LaTeX code while working, but it's there if you need detailed customisations.

Google Docs is the word processor that makes up part of Google Drive, meaning that it's the ultimate when it comes to accessing over the cloud. It also has some very nice features such as an extremely up-to-date grammar and spellchecker.

CREDIT: Getty Images/Sean Gladwell

Spelling and grammar checker

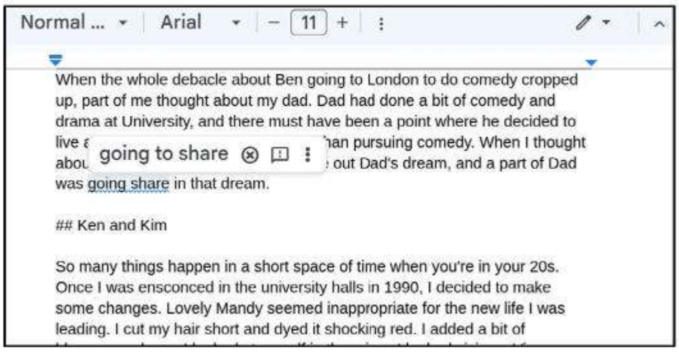
All of the solutions have a spellcheck, but the quality varies.

Google Docs's* spellchecker is impressive, particularly if you work in an area like technology writing as it always seems to be up to date. For example, it recognises words like 'Nvidia' and 'rasterisation'. Medical science terms such as 'Neuroectoderm', 'parvocellular' and 'neurosecretory' were all recognised, too. In cases where it's almost bound to be a typo, it autocorrects, and we feel comfortable leaving that enabled.

The checking for grammatical mistakes is also excellent. It picks up things like the infamous 'the the' typo and mistakes like 'John and Harry is going to the beach'. It can spot mistakes such as 'effect' instead of 'affect' or 'markup' rather than 'mark up' from the context. However, it doesn't offer any advice on writing style, as some grammar checkers can.

Impressively, *LibreOffice* did quite well in this regard, even though it isn't able to use the same level of cloud-based magic to keep up with specialist terminology. It recognised some medical terms such as 'neurosecretory' and 'melatonin' but not as many as *Google Docs*. It doesn't include a grammar checker at all.

Purely in terms of spellchecking, *Visual Studio Code* with the LanguageTool extension is at about the same level as *LibreOffice* when it comes to recognising modern and specialist terms, and company names. The advantage is that it's a grammar checker, too. It's not quite on the same level as the *Google Docs* one, but it has explanations, which *Google Docs* doesn't, and some style tips, such as if a sentence is too long or you have started a sentence with the same word too many times. It sometimes generates false positives, particularly in 'picky' mode.



When the whole debacle about Ben going to London to do comedy cropped up, part of me thought about my dad. Dad had done a bit of comedy and drama at University, and there must have been a point where he decided to live a han pursuing comedy. When I thought abou out Dad's dream, and a part of Dad was going share in that dream.

going to share

Ken and Kim

So many things happen in a short space of time when you're in your 20s. Once I was ensconced in the university halls in 1990, I decided to make some changes. Lovely Mandy seemed inappropriate for the new life I was leading. I cut my hair short and dyed it shocking red. I added a bit of

As well as spelling mistakes, Google Docs can spot grammatical errors. These are often as simple as accidentally using the same word twice or forgetting a comma.

LyX has the basic spellchecking features that you'd expect. It works, but the dictionary is rather old fashioned and fails to recognise many contemporary company names (Nvidia, for example) and technical terms. It has no grammar checker at all.

FocusWriter has standard spellcheck facilities. It understood the company name 'Nvidia' and technical and scientific terms. It doesn't have any grammar checking or suggestions, which would have been nice to have on a fiction writer's tool like this.

VERDICT

VISUAL STUDIO CODE	8/10	GOOGLE DOCS	10/10
LIBREOFFICE WRITER	6/10	FOCUSWRITER	6/10
LYX	5/10		

Google Docs has the most sophisticated spellchecker we've seen, and its grammar checker is extremely useful.

Handling images

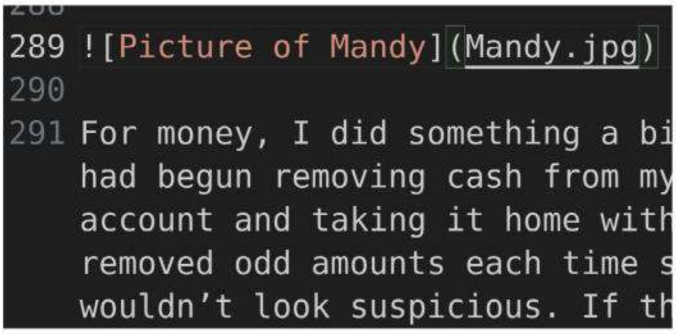
Charts, diagrams, photographs, maps – some books require images.

L*yX's* facilities for dealing with images (such as charts and photographs) are excellent. You can place images into a floating image frame that is placed in the document when it is compiled. This also means that elements such as charts are always the latest version. You can refer to a figure and see it referenced as "see figure 4 on this page". It's also possible to include more than one image within a frame. Support for any image type can be added to *LyX* in the Converters section of its preferences if you have a command-line tool for processing it.

Markdown in *Visual Studio Code* has the basic facilities for working with images, but there aren't many advanced features, and actually adding the images involves manual work.

FocusWriter has no facilities for handling images at all.

LibreOffice has comprehensive image features in terms of adding things like labels and borders, and it's possible to number figures and include them in a contents list. Of the two, *LyX* can possibly take things further as you can customise the underlying LaTeX code (if you know LaTeX) to do nearly anything, but it's nowhere near as intuitive as *LibreOffice*. *Draw*, a vector drawing, charting and diagram creation app, is part of the suite.



```
![Picture of Mandy](Mandy.jpg)

For money, I did something a bi
had begun removing cash from my
account and taking it home with
removed odd amounts each time s
wouldn't look suspicious. If th
```

It's possible to add images to a Markdown file through tags. This is OK for occasional images, but it would become a chore if you had to add a lot of them.

Google Docs's image features should cover the basic requirements that most people are likely to need. It's heavily connected to the cloud at all times, so you have the ability to import images from sources such as your Google Drive account, the web or a webcam.

VERDICT

VISUAL STUDIO CODE	5/10	GOOGLE DOCS	7/10
LIBREOFFICE WRITER	8/10	FOCUSWRITER	N/A
LYX	8/10		

LyX has impressive features and scope for customisation; however, LibreOffice is easier to use in this area.

User interface and experience

A pleasant writing environment.

There are two basic approaches to how an application handles the text of a book. Some programs are WYSIWYG (What You See Is What You Get) and faithfully reflect the exact appearance of the finished document while it is being created. However, WYSIWYG is often distracting and inconvenient when working on a book project. For example, you might prefer to work with a sans serif font on screen even though a serif font is more usual for a printed book.

This separation between the text editing phase and the eventual output is particularly relevant these days, as you could well be targeting a variety of output formats, such as the web, multiple ebook formats and a traditional printed book.

When we're working on a book project, we basically just want to get on with the writing. If it's a project where we have to get our hands dirty with non-textual content such as tables and figures, we want to be able to add those things easily.

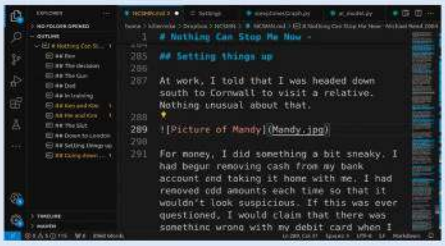
Visual Studio Code 7/10

Using Markdown in *Visual Studio Code* sounds like a technical solution, but in day-to-day use, it's free of distractions. As this is markup, there is a separation between working on the text of your book and the eventual formatting. The text is one long page and the default colour scheme uses a dark grey background with a light grey font.

Along the side, a scrollbar includes a graphical overview of the text with highlighted grammar and spelling errors.

Nearly everything's easily accessible from the keyboard, and everything's super-fast. If you're writing a novel-type book, you won't have to do much formatting for the actual writing, and the Markdown syntax is easy to learn. If you're writing a book that requires a lot of images, the complexity of hand-coding markup naturally increases.

If you are used to text editors, this could be the most relaxing setup for a book.



```
# Nothing Can Stop Me Now -

## Setting things up

At work, I told that I was headed down
south to Cornwall to visit a relative.
Nothing unusual about that.

![Picture of Mandy](Mandy.jpg)

For money, I did something a bit sneaky. I
had begun removing cash from my bank
account and taking it home with me. I had
removed odd amounts each time so that it
wouldn't look suspicious. If this was ever
questioned, I would claim that there was
something wrong with my debit card when I
```

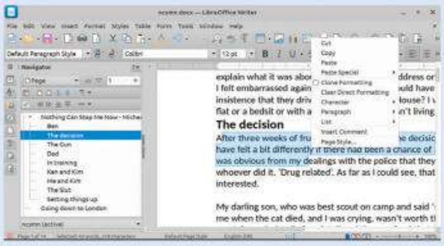
LibreOffice Writer 9/10

In most respects, *LibreOffice Writer* uses the layout of a traditional word processor. In day-to-day use, it gives you a perfect mix of pull-down menus and toolbars alongside keyboard shortcuts. It can even switch from toolbar to ribbon-style interface, if that's what you prefer.

You can have multiple documents open, but they are in their own windows rather than multiple tabs, which is a shame because many writers like to have notes in a separate document.

There is some detachment between the editing environment and the finished document thanks to its web view mode, which means that the text fills the window and exists on one long page. But there's a reason most programmer's editors prefer a dark background and a sans serif font, which is difficult to achieve in *LibreOffice*.

LibreOffice Writer offers a mixture of the classic interface mixed with some modern refinements.



explain what it was abo… …ddress or
I felt embarrassed again… …uld have
insistence that they driv… …louse? I
flat or a bedsit or with a… …n't living

The decision

After three weeks of fru… …e decisic
have felt a bit differently if there had been a chance of
was obvious from my dealings with the police that they
whoever did it. 'Drug related'. As far as I could see, that
interested.

My darling son, who was best scout on camp and said
me when the cat died, and I was crying, wasn't worth t

For fiction writing

For prose, a smooth workflow is more important than advanced features.

When configured for writing, the *Visual Studio Code* interface is plain and distraction-free. In most cases, you can operate everything from the keyboard, including using the inline spell/grammar check. We're not going to reduce the score due to Markdown being a markup language, as typing `##` before a chapter title isn't much of a hardship.

Google Docs's pageless mode is welcome, but you don't have to use it if you prefer to work with pages as a progress marker. Its excellent grammar check constantly points out common mistakes, freeing you to get on with your creative flow.

LyX is pleasant to work with for a fiction book, as you need never see any markup code on a day-to-day basis. It's a shame it doesn't have an inline grammar checker like some of the other options. It does, however, have a full-screen mode with a minimalist UI, for people who like to work like that, along with complete freedom of font and colour scheme while editing.

LibreOffice offers a classic word processor environment. The only snag with that is that it means that there is never a complete detachment between your layout choices while working and when outputting to formats such as PDF, meaning that you might not have your ideal font and colour scheme on screen while working.

This is where *FocusWriter* comes into its own, because it is primarily designed for fiction writing. Apart from a mellow, distraction-free writing environment, daily goals and a daily progress record are available. As with *LyX* and *Visual Studio Code,* it's easy to select a theme and font choice that are different from those of the eventual output file.

VERDICT

VISUAL STUDIO CODE	8/10	GOOGLE DOCS	9/10
LIBREOFFICE WRITER	7/10	FOCUSWRITER	8/10
LYX	8/10		

Things are fairly even, but Google Docs has its grammar checker and cloud basis in its favour when working on fiction.

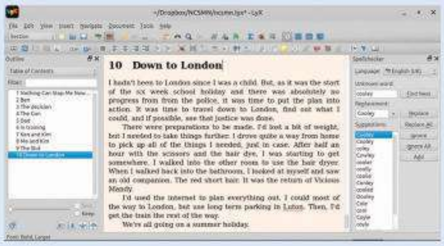
LyX 9/10

LyX sits halfway between a text editor and a word processor. It offers separation between the writing environment and the finished document. For example, if you resize the window, the text reflows to fit, and text can be justified or left aligned. It defaults to a pinkish pastel background with black text, and the text sits on a single, continuous page. The editing font can be different from the eventual output.

Working with *LyX* is different from working with a markup language because font weights are reflected on screen for things like chapter headings. Images are also shown as part of the document, but elements like that can be folded away.

There's a small status bar at the bottom of the window and the keyboard shortcuts are displayed here after you've carried out an action – a nice touch.

Setting up *LyX* can be a bit complicated in parts, but for actual writing, it's as distraction-free as possible.



10 Down to London

I hadn't been to London since I was a child. But, as it was the start of the six week school holiday and there was absolutely no progress from from the police, it was time to put the plan into action. It was time to travel down to London, find out what I could, and if possible, see that justice was done.

There were preparations to be made. I'd lost a bit of weight, but I needed to take things further. I drove quite a way from home to pick up all of the things I needed, just in case. After half an hour with the scissors and the hair dye, I was starting to get somewhere. I walked into the other room to use the hair dryer. When I walked back into the bathroom, I looked at myself and saw an old companion. The red short hair. It was the return of Vicious Mandy.

I'd used the internet to plan everything out. I could most of the way to London, but use long term parking in Luton. Then, I'd get the train the rest of the way.

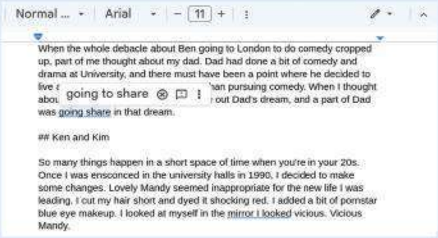
We're all going on a summer holiday.

Google Docs 7/10

In many respects, the *Google Docs* GUI resembles a classic word processor interface, giving it an advantage of familiarity. Being WYSIWYG is a double-edged sword when it comes to writing a book. Although you are always aware of the exact layout, this can be distracting when you're working on inputting the text.

Although it sticks fairly close to traditional word processor conventions, it's possible to work with one continuous page for the whole document. We didn't have any problems operating most of the program from the keyboard while writing, and things like starting a new heading can be done from keyboard combos.

No matter how fast your computer, there's always a slight feeling of delay for a program that uses web technologies once you're using the mouse. In particular, things such as double-clicking on text and accurate highlighting with the mouse can take more than one attempt to get right.



When the whole debacle about Ben going to London to do comedy cropped up, part of me thought about my dad. Dad had done a bit of comedy and drama at University, and there must have been a point where he decided to live a ... han pursuing comedy. When I thought abou ... out Dad's dream, and a part of Dad was going share in that dream.

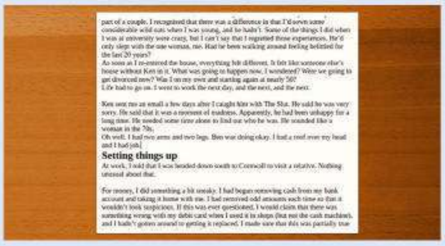
Ken and Kim

So many things happen in a short space of time when you're in your 20s. Once I was ensconced in the university halls in 1990, I decided to make some changes. Lovely Mandy seemed inappropriate for the new life I was leading. I cut my hair short and dyed it shocking red. I added a bit of pornstar blue eye makeup. I looked at myself in the mirror I looked vicious. Vicious Mandy.

FocusWriter 6/10

When you first launch *FocusWriter*, all you see is a blinking cursor on a white page upon a wood-textured backdrop, which takes up the full screen. Talk about minimalism! If you move the pointer to the top, a fairly conventional pull-down menu appears. Similarly, move the pointer to the bottom of the screen, and a status bar appears offering a conventional word count along with more specialist facilities

The text editing page is a continuous one, but *FocusWriter* can optionally display a page count in the status area. The status area, the menu and the toolbar can be set to be permanently on display. There is a detachment between the output layout and theme and the on-screen one. For instance, we liked the green on black, hacker-style theme with the monospaced font for actual writing work. The environment feels like a mixture of a word processor and a text editor that's optimised for fiction writing.



part of a couple. I recognised that there was a difference in that I'd sown some considerable wild oats when I was young, and he hadn't. Some of the things I did when I was at university were crazy, but I can't say that I regretted those experiences. He'd only slept with the one woman, me. Had he been walking around feeling belittled for the last 20 years?
As soon as I re-entered the house, everything felt different. It felt like someone else's house without Ken in it. What was going to happen now, I wondered? Were we going to get divorced now? Was I on my own and starting again at nearly 50?
Life had to go on. I went to work the next day, and the next, and the next.

Ken sent me an email a few days after I caught him with The Slut. He said he was very sorry. He said that it was a moment of madness. Apparently, he had been unhappy for a long time. He needed some time alone to find out who he was. He sounded like a woman in the 70s.
Oh well. I had two arms and two legs. Ben was doing okay. I had a roof over my head and I had job.

Setting things up

At work, I told that I was headed down south to Cornwall to visit a relative. Nothing unusual about that.

For money, I did something a bit sneaky. I had began removing cash from my bank account and taking it home with me. I had removed odd amounts each time so that it wouldn't look suspicious. If this was ever questioned, I would claim that there was something wrong with my debit card when I used it in shops (but not the cash machine), and I hadn't gotten around to getting it replaced. I made sure that this was partially true

For non-fiction books

Things get more complicated here as extra features come into play.

LyX is in its element when creating a long, non-fiction document with an academic bent. Its styles system can be customised for practically any arrangement that you need, particularly if you're prepared to delve into some LaTeX. If you're a mathematician, it's possible to add complex equations into the document. The facilities for working with inline citations from a database are comprehensive and built in. You can also integrate the output of scripts and command-line tools into the document. Because the layout is detached from the editing style, you can do things such as fold away elements such as images and notes while you're working.

Visual Studio Code has a natural advantage when working with code, making it a good choice for non-fiction projects about programming. However, if you're writing a book that involves lots of images, tables and citations, any hand-written markup needs an extra layer of work, a disadvantage of *Visual Studio Code*.

LibreOffice has an advantage in that it's part of a suite of office tools that includes applications such as a database and a spreadsheet, invaluable for a lot of non-fiction projects, and the same can be said for *Google Docs*. When it comes to things such as charts and tables, *LibreOffice* has good facilities for including them, and much of this material can be created in the other parts of the overall suite. *Google Docs* shares this advantage. When creating a typical non-fiction book, *LibreOffice* and *Google Docs* are easier to set up than *LyX*.

As we've already mentioned, *FocusWriter* is orientated towards fiction writing, and it would be difficult to use it for a more complex book project.

VERDICT

VISUAL STUDIO CODE	5/10	GOOGLE DOCS	7/10
LIBREOFFICE WRITER	8/10	FOCUSWRITER	4/10
LYX	9/10		

LyX is the ideal choice for academic writing, but Google Docs and LibreOffice are also good choices.

Output formats

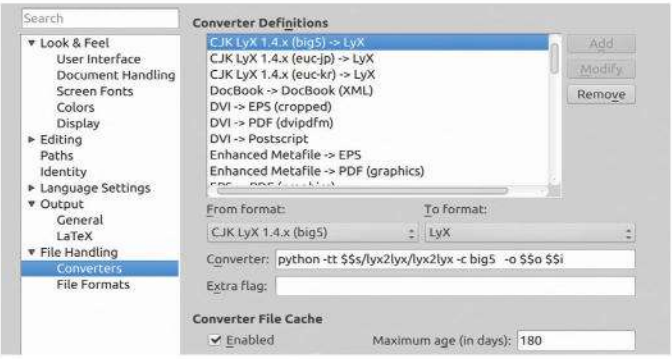
Converting your book into a publishable format.

LyX sometimes requires a bit of configuration and installation of dependencies, but once it is working, the sky's the limit in terms of the formats and standards it can handle. The underlying LaTeX code is an ideal starting point to produce clean output to other file formats.

Google Docs can output to most of the popular publishing formats such as ePub and PDF, but it doesn't produce particularly clean output, evident if you examine its dreadful-looking HTML output in a text editor. It can output to ePub, but the layout was unreadable in *Calibre's* ebook reader. It's a bit more trustworthy when outputting to PDF, and you could probably write, design and publish a book within *Google Docs* if this was the end format.

There were some dependency issues that needed sorting out before *Visual Studio Code* could reliably output to ePub and PDF. Books in the latter format were OK-looking, but the ePub file looked rather wonky in a reader. We'd probably use an external tool such as *Calibre* to make the files ourselves.

LibreOffice can also output your book to both ePub and PDF formats, and the facilities worked very well. Importantly, there is a pop-up dialog box that enables you to specify the metadata in both cases. The ePub file looked great in the reader in which we tried it. Because PDF is a business format, we would trust the files to be robust.



Converter Definitions

- CJK LyX 1.4.x (big5) -> LyX
- CJK LyX 1.4.x (euc-jp) -> LyX
- CJK LyX 1.4.x (euc-kr) -> LyX
- DocBook -> DocBook (XML)
- DVI -> EPS (cropped)
- DVI -> PDF (dvipdfm)
- DVI -> Postscript
- Enhanced Metafile -> EPS
- Enhanced Metafile -> PDF (graphics)

From format: CJK LyX 1.4.x (big5)

To format: LyX

Converter: python -tt $$s/lyx2lyx/lyx2lyx -c big5 -o $$o $$i

Extra flag:

Converter File Cache

☑ Enabled

Maximum age (in days): 180

Any format that can be converted on the command line can be supported in LyX. It is a bit technical to get it all working, but it's highly flexible.

The limited output formats of *FocusWriter* suggest you'll have to use another app to create the final production files that can be turned into a finished book. This may have been the developer's best strategy, rather than adding half-hearted features.

VERDICT

VISUAL STUDIO CODE	5/10	GOOGLE DOCS	7/10
LIBREOFFICE WRITER	9/10	FOCUSWRITER	5/10
LYX	9/10		

Once configured, LyX can handle any output format. LibreOffice's ePub and PDF look good. For the others, we'd use an external tool.

Cloud and collaboration

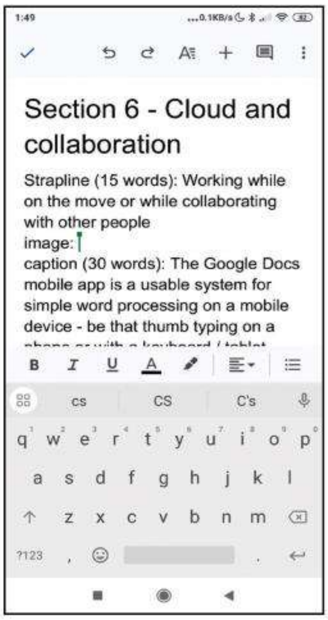
Working while on the move or collaborating with other people.

Documents in *LyX* are text files with markup tags, so you can put together whatever backup and networking system you like, but you have to do it yourself. However, it doesn't come with any collaboration features, and that's a bit of a weakness. It's worth taking note that, although *LyX* documents can be opened in a text editor, *LyX* files aren't very well known, and most people won't know what to do with them.

Google Docs was created to work in the cloud, and this is where it's in its element. Without having to manually save, you can walk from one computer to the next and keep working from where you left off. Its excellent collaboration features highlight the parts that other people have added, and you can create rules to control who can view or edit your documents. You can work offline for a while, but you can't work off-grid in the long term.

The *Google Docs* mobile app is perfectly usable, and you can set up a simple, pocketable word processing environment with a Bluetooth keyboard and phone. You can always input some notes with thumb typing, but how much actual work you could do on a novel like that is debatable. However, with a tablet and keyboard, this could be a perfectly capable setup for professional work.

Visual Studio Code and Markdown are a better fit to integrate with a system such as Git or CVS than the other choices. From the get-go, *Visual Studio Code* has some versioning and history features set up. With planning, you could perhaps set something up so that you can access the Markdown files (which are simply text files) on a mobile device.



Section 6 - Cloud and collaboration

Strapline (15 words): Working while on the move or while collaborating with other people
image:
caption (30 words): The Google Docs mobile app is a usable system for simple word processing on a mobile device - be that thumb typing on a

The Google Docs mobile app is a usable system for simple word processing on a mobile device – be that thumb-typing on a phone or with a keyboard/tablet combo.

LibreOffice is a let-down when it comes to access over the internet and sharing in the cloud. As with any system, there are things you could do to get at your documents while on the move, but it's not baked in. *Collabora Online* is a third-party freemium service that uses *LibreOffice* tech in the cloud.

Finally, *FocusWriter* doesn't have any collaboration or cloud access features.

VERDICT

VISUAL STUDIO CODE	7/10	GOOGLE DOCS	9/10
LIBREOFFICE WRITER	5/10	FOCUSWRITER	3/10
LYX	5/10		

Google Docs is fantastic in the cloud, but not so good offline. Visual Studio Code could be jury-rigged for sharing.

The verdict

Editors for authors

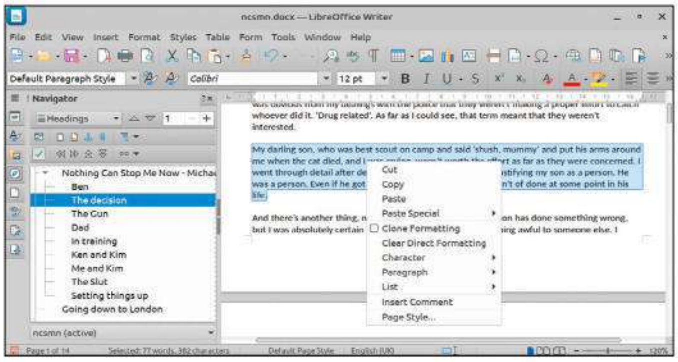
LibreOffice is an amazing piece of software, and it's the best all-rounder in our opinion. It models itself on the traditional word processor, meaning it may not be the boldest choice. It has an interface that is familiar to the majority of people, with practically no software configuration required before you can get working on your book. There are some features for detaching the editing layout from the output layout. Beyond this, it offers a surprising amount of detail in areas such as contents and index tables, and the finer points of presenting and labelling images.

Google Docs is a great system for working with a document such as a book. For non-fiction, it covers the basics in every area, such as adding citations, along with charts and tables. For any type of writing, it has an intelligent grammar checker and spellchecker that is kept constantly updated. It's the best choice for working while on the move or collaborating with other contributors.

LyX is an excellent choice for working with book-length projects. There's a complete detachment between the eventual layout and the environment that you set up for yourself while doing the day-to-day writing work. If you're writing a novel, you need only get into the technical side of setting up the layout once the project is nearing completion, if at all. If you're working on a non-fiction book, you'll love the advanced features for tables, charts, photos and citations.

Markdown within *Visual Studio Code* offers a simple, distraction-free environment for working with long text documents. Once we'd added a few extensions, such as the LanguageTool grammar and spelling checker and a word count plugin, we were amazed at how few problems we ran into when working with a book-length project. Working this way, you can use the normal facilities for collaboration and version control as you would when working with code in *Visual Studio Code*. We'd be reluctant to use it for a non-fiction book project, as day-to-day workflow can get quite complicated when working with elements such as images, tables and citations.

FocusWriter is a great piece of software that does one thing really well. It gives you a blank page, spellcheck and some other facilities, such as goals and daily targets, and it's all wrapped up in a minimalist environment.



whoever did it. 'Drug related'. As far as I could see, that term meant that they weren't interested.

My darling son, who was best scout on camp and said 'shush, mummy' and put his arms around me when the cat died, and I ... rt as far as they were concerned. I went through detail after de... stifying my son as a person. He was a person. Even if he got ... n't of done at some point in his life.

And there's another thing, n... on has done something wrong, but I was absolutely certain ... ing awful to someone else. I

1st LibreOffice 9/10

Web: **www.libreoffice.org**
Licence: **Mozilla Public License (MPL)** **Version:** **MPL-2.0**
Brilliant all-rounder with a classic design. Feature-heavy, but easy to use.

2nd Google Docs 8/10

Web: **https://docs.google.com**
Licence: **Proprietary** **Version:** **Rolling updates**
Born to work in the cloud. Excellent grammar checker and spellcheck.

3rd LyX 8/10

Web: **www.lyx.org**
Licence: **GPL 2.0 or later** **Version:** **2.3.7**
Particularly good for academic writing. Customisable and flexible.

4th Visual Studio Code 7/10

Web: **https://code.visualstudio.com**
Licence: **MIT** **Version:** **1.88.0**
DIY solution. Surprisingly good for fiction. Appealing to programmers.

5th FocusWriter 7/10

Web: **https://gottcode.org/focuswriter**
Licence: **GPL 3** **Version:** **1.8.6**
Minimalist environment for fiction writers. Does one thing very well.

» ALSO CONSIDER

We've covered *LyX* (an editor that runs on top of LaTeX) but it is possible to manually create LaTeX code in a text editor such as *Visual Studio Code*. Pure LaTeX is common in academic and scientific circles, and like *LyX*, it produces textbook-like results with ease. We didn't cover that because we chose Markdown as the text editor/markup combination to examine.

There are Linux-compatible office suites other than *LibreOffice*, such as *OnlyOffice*, *WPS Office* and *FreeOffice*, and all of these could handle a book. Cloud operation is weak for *LibreOffice*, but *OnlyOffice* can be hosted on a remote server as part of a paid plan or your own locally hosted server.

WordGrinder is a minimalist word processor in the form of a text-mode console application. It has basic support for styling text, a simple spellchecker and not much else. If you're heavily interested in text mode, consider *Org Novelist*, the novel-writing add-on for text editor *Emacs*. LXF

Ultimate desktop upgrade

Transform your desktop with hot KDE Plasma and a little **Jonni Bidwell**.

If you've never heard of KDE, you're missing out on a fantastic desktop experience. Or maybe you have heard of it, but recall it bringing your system to a crawl in the 2010s. Rest assured those days are gone and despite looking beautiful, KDE Plasma (to give the desktop its proper title) is quite light on resource usage.

Trying out the latest KDE Plasma offerings used to be tricky. You'd generally have to wait until the release cadence allowed for it to be included in a major distribution. Even then you'd run into packaging or compatibility issues. But thanks to KDE Neon, this is no longer the case. KDE Neon is based on Ubuntu, but includes the latest KDE workings, so you get all the newest desktop offerings on a rock-solid foundation. We show you how to install Neon and get the most out of it.

We've also got tips on how to install KDE Plasma alongside your current desktop, so you can switch between what you know and what you want to know. We cover everything the tweakiest of desktop tweakers will want to (er) tweak, too, and explain how to use the *KDE Connect* app on any distro, so you can get your PC and your mobile devices working in harmony.

CREDIT: Magictorch

Enter the K desktop environment and apps

Introducing KDE Plasma 6.0 and the surrounding terminology.

Anyone who can name two Linux desktop environments off the top of their heads (admittedly a small sample of the general population) will very likely say "Gnome and KDE". This duality was even brought up in *Mr Robot.* But, at least for the last decade or so, it seems as though KDE Plasma (to give it its correct title) has always played second fiddle to Gnome. This is perhaps unfair. Lots of other desktop environments exist, and most of these are based on *GTK* (the toolkit that underlies Gnome). Over the last decade, *GTK* has become inextricably linked *[that is not a good pun – ed]* to certain Gnome libraries. So, by journalistic oversimplification, we might get away with saying that lots of these other desktops (MATE, Budgie, Xfce, Cinnamon and even the reboot of Unity) incorporate a significant portion of Gnome plumbing, even if they don't look like it.

KDE Plasma uses the *Qt* (often pronounced "cute") toolkit, which has vastly different origins (we'll get into the ins and outs of the two toolkits and related nomenclature later; do bear with us). As far as desktop use goes, *Qt* is only used in KDE Plasma and LXQt (the lightweight desktop used in Lubuntu). In contrast to *GTK*, *Qt* has become much very much decoupled from the rest of the KDE ecosystem. This (specifically the strict delineation of *Qt*, KDE Frameworks and the KDE Applications themselves) was one of the major achievements of KDE Plasma 5, which we wrote about back in **LXF206**. There we noted the desktop was slicker, less bloated and much more welcoming to newcomers than its predecessor (which by the end of its tenure was called KDE SC4).

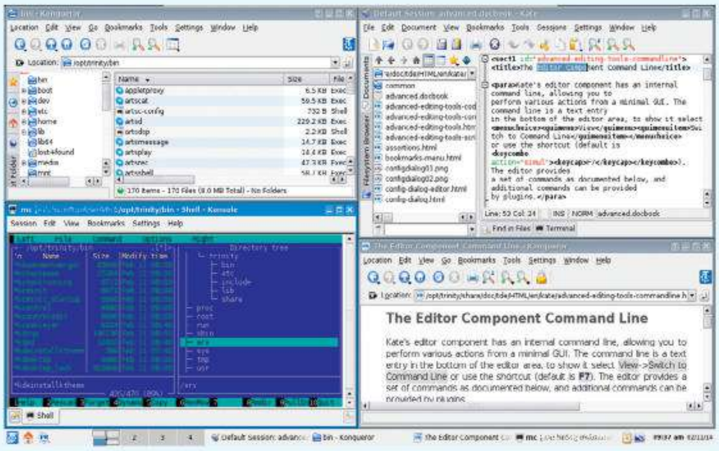
In February, KDE Plasma 6.0 was released, and we have been itching to write about it ever since. But we figured it diplomatic to wait until the bugs were ironed out first. So here we are, writing just after KDE 6.0.4 has been released, scratching that itch. Those who remember the (lengthy and arduous) transition from KDE 4 to KDE Plasma 5 will surely recall that the two releases were cosmetically (and also constitutionally) very different. We'll clear one thing up right now to avoid disappointment: KDE Plasma 6 does not look hugely different from its predecessor. There aren't really any new flashy user-facing features. Indeed, a lot of the changes are incremental continuations of the structure introduced in KDE Plasma 5. That notwithstanding, there are some major under-the-hood changes and an all-round great desktop. Over the page we show you KDE Neon, the easiest way to sample this latest effort.

» KDE LORE

The first release of KDE happened in 1998, back when "year of the Linux desktop" referred to the fact that it took you about a year to get a working Linux desktop. Back then the acronym stood for Kool Desktop Environment, and was also a tongue-in-cheek jab at the Common Desktop Environment (CDE) used by most commercial Unix-es at the time.

We're old enough to remember using KDE 2 at the beginning of the millennium, which introduced the many-faced *Konqueror* (at once a file manager, web browser and document viewer), as well as many other applications and libraries beginning with K. But it was its successor that really saw KDE going mainstream. KDE 3 distinguished itself from Gnome 2 by being more Windows-like. It had a post-Windows 95-style Applications (Start) menu and a fetching blue theme (so former XP users seeking refuge would feel a little more at home). Gnome 2 was by comparison more akin to Mac OS (as it was then capitalised), with its top menu bar and the immutable cascading Applications menu therein.

KDE 4 was like Marmite. You either loved it or hated it. It was nothing if not configurable, but it shipped with lots of desktop effects turned on by default. Which was no use if you didn't have a fancy graphics card, or if its drivers weren't set up just right. A common criticism was that it was "too configurable". Users struggled to find exactly where to turn off these effects, or to undo changes they accidentally made (such as inadvertently removing the main panel). Gnome users (and users of other bare-bones desktops) also complained that you couldn't install (even simple) KDE applications without pulling in great swathes of the desktop as dependencies.



```
<sect1 id="advanced-editing-tools-commandline">
<title>The Editor Component Command Line</title>

<para>Kate's editor component has an internal
command line, allowing you to
perform various actions from a minimal GUI. The
command line is a text entry
in the bottom of the editor area, to show it select
<menuchoice><guimenu>View</guimenu><guimenuitem>Switch to Command Line</guimenuitem></menuchoice>
or use the shortcut (default is
<keycombo
action="simul"><keycap>F7</keycap></keycombo>).
The editor provides
a set of commands as documented below, and
additional commands can be provided
by plugins.</para>
```

The Editor Component Command Line

Kate's editor component has an internal command line, allowing you to perform various actions from a minimal GUI. The command line is a text entry in the bottom of the editor area, to show it select View->Switch to Command Line or use the shortcut (default is **F7**). The editor provides a set of commands as documented below, and additional commands can be provided by plugins.

Location Edit View Go Bookmarks Tools Settings Window Help
File Edit Document View Bookmarks Tools Sessions Settings Window Help
Location: /opt/trinity/bin
Location: /opt/trinity/share/doc/tde/HTML/en/kate/advanced-editing-tools-commandline.h
Line: 53 Col: 24 INS NORM advanced.docbook
Find in Files Terminal
Documents
Filesystem Browser
Shell

The Trinity Project (https://trinitydesktop.org) is keeping the spirit of KDE 3 alive.

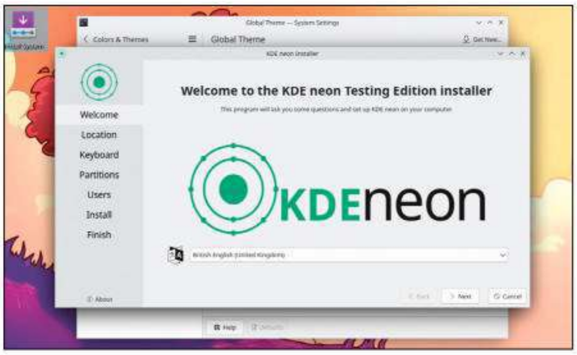
Introducing KDE Plasma 6.0

Get started with one of the most powerful desktops in the world.

Fedora has long been a bastion of the Gnome desktop, and it's been used in Ubuntu since 2017. Pop!_OS's COSMIC is packaged as a Gnome shell extension (although the next version will be a totally separate affair). And SolusOS's Budgie desktop is based on Gnome and retains its look and feel. By comparison, no mainstream distro offers KDE Plasma by default. Sure, desktop-agnostic affairs such as Debian and OpenSUSE all let you select it from the installer. But in terms of distros that ship it (apart from Kubuntu, the Fedora KDE respin and Manjaro KDE) by default, you've got, er, Mageia and KaOS. All fine, don't get us wrong, but not really ones that grab headlines.

Of course, (with determination) you can install KDE on any distribution you like. But it's not something that beginners, or anyone who wants to use their computer rather than mess with it, will necessarily want to do. So, to address this dearth of KDE-focused distros, the team launched KDE Neon in 2021. It uses Ubuntu LTS as a base, but strips out the Gnome desktop and all the Gnome apps. In their stead, we find the very latest KDE Plasma desktop, with the cream of the KDE application suite. Originally, KDE Neon didn't describe itself as a distro, because there weren't enough KDE applications to satisfy most people's expectation of a 'real' distro. But now there are (or perhaps Ubuntu's decision to slim down its bundled selection lowered the bar).

You can read all about KDE Neon at **https://neon.kde.org**, but we're going to focus on installing it. Follow the download link and you're offered four editions: User, Testing, Unstable and Developer. We recommend you start with the User flavour. The Testing edition has some newer components and has undergone some bug testing, so will probably not hurt. And as you might guess (and quoth the website "will contain many bugs") Unstable is on the bleeding edge. The Developer edition is the Unstable edition plus everything you need to contribute to KDE Neon or the wider KDE ecosystem. To clarify, "KDE" refers to that ecosystem and the community around KDE Plasma. The desktop is KDE Plasma, and you often see lazy journalists referring to it as just KDE, which is wrong. If you want to split hairs, the desktop should be called KDE Plasma Desktop, as there is a mobile edition, too, but we feel this digression has been more trouble than it's worth.



Welcome to the KDE neon Testing Edition installer

This program will ask you some questions and set up KDE neon on your computer.

KDE Neon's Calamares-powered installer and our in-box guidance will have you running Plasma 6 in no time.

Having chosen your edition, write it to a USB device (you'll find links to KDE's own *Imagewriter* for Linux or Windows in the Download section). You might prefer to use KDE Neon in a virtual machine (VM), in which case follow the instructions and turn on 3D acceleration in the guest. This means ticking a box in *VirtualBox*, or selecting VirtIO video and then ticking a box (in the Spice Graphics section) in *Virtual Machine Manager*. Boot the USB drive (we assume you're familiar with this dance) and you're greeted with a welcome screen.

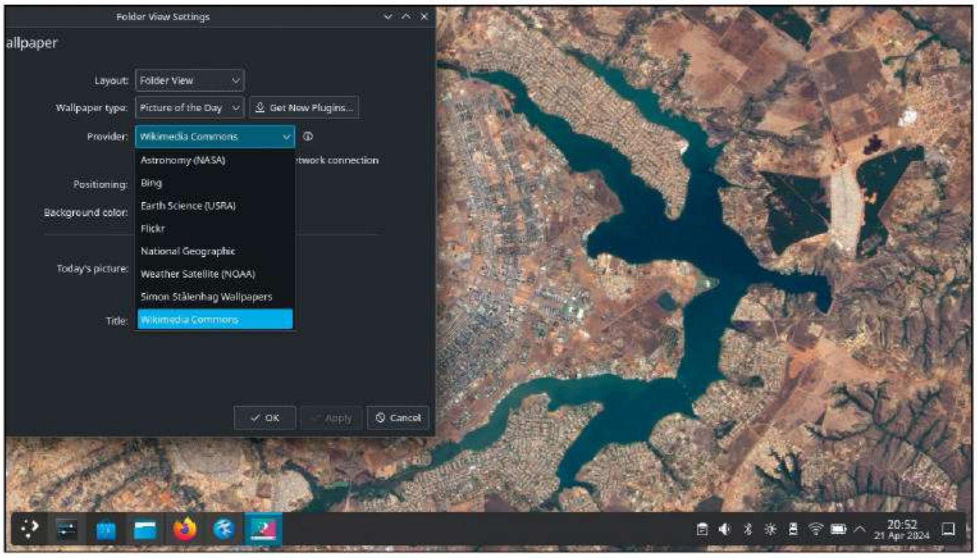
From here you can go ahead and install KDE Neon, but we recommend hitting Skip first and exploring the

» INSTALLING NEON

When ready, double-click the Install System icon on the desktop. You're asked the usual localisation questions before moving on to the partitioning screen. Neon can happily install alongside other OSes (if space), resizing their partitions as required, which is fine for flavours of Linux, but we caution against resizing a Windows partition. If you really must, it's better to use Windows' own *Disk Management* tool, then return to Neon and install in the newly liberated space. Ideally, you would install Neon on a separate drive from Windows.

If you have lots of flavours of Linux installed, and one you don't like (or that is broken beyond repair), the installer can use that OS's partition and leave the rest alone. KDE Neon requires 10GB, but we recommend allowing much more. Once you've sorted that, choose a username, password and name for your machine. Then hit Install and make a cup of tea or peruse the welcome slideshow (introduced by Konqi, the KDE mascot).

live environment. This way you know what you're letting yourself in for. From the get-go, you'll see that KDE Plasma is a traditional-looking desktop. The icon in the lower-left opens an Applications menu and any apps you open appear in the adjacent traditional-looking task bar. A trio of apps are pinned next to the Applications menu for quick access. These are the *Discover* software centre, *Dolphin* file manager and *Firefox* web browser. To the right of the task bar is an orthodox system tray, notifications area, and a date/time widget. If you've tried Gnome and not liked its hypermodern stylings, this might all sound like a breath of fresh air. Or if you're open-minded, maybe you'll enjoy the merits of both these desktop environments.



Folder View Settings

allpaper

Layout: Folder View

Wallpaper type: Picture of the Day

Get New Plugins...

Provider: Wikimedia Commons

- Astronomy (NASA)
- Bing
- Earth Science (USRA)
- Flickr
- National Geographic
- Weather Satellite (NOAA)
- Simon Stålenhag Wallpapers
- Wikimedia Commons

twork connection

Positioning:

Background color:

Today's picture:

Title:

✓ OK | Apply | Cancel

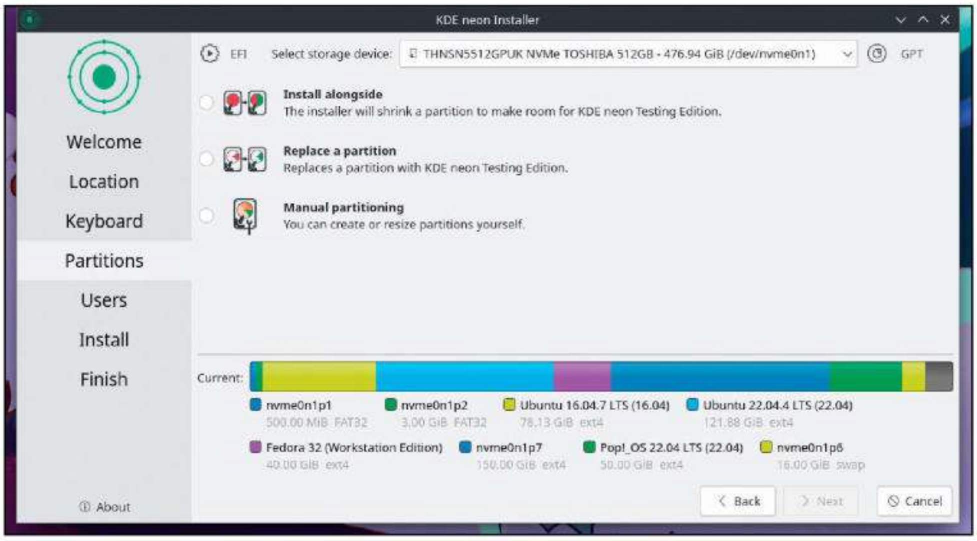
20:52 21 Apr 2024

Plasma includes a variety of picture-of-the-day plugins, from the likes of Bing, NASA or Wikimedia Commons (pictured).

Open *Dolphin* and you'll find its clean-looking interface belies a very powerful file manager. Most options are available from the 'hamburger' (☰) menu, top-right. You might be shocked at the lack of menu bar, but you can summon it by pressing Ctrl+M, which also disappears the hamburger icon. This is true for a lot of KDE apps, as old-school menu bars are often seen today as a clunky way of getting things done. If you don't like this, just remember that in Gnome the maxim "hamburger uber alles" seems to have taken root, with no native Gnome apps having 'proper' menus. If you push F4 (or go to the Panels section of the hamburger menu), a terminal pops up at the bottom. This is great for doing hybrid file management, since the current directory in the terminal follows you as you navigate the folders in *Dolphin*. Even for virtual filesystems such as Trash and network shares.

Kustomising KDE

You might also notice the drop shadow around the *Dolphin* window. You might think this is stylish, or too shady, or a downright waste of resources. It can be changed from the System Settings (inside the Settings menu of the Applications menu). In the menu on the left, navigate to Colors And Themes in the Appearance & Style section. Then click on the (easily missed) Edit button in the lower-right of the Breeze theme icon. You'll find all the options you need in the Shadows And Outline tab. In contrast to Gnome, KDE is highly configurable (without the need for third-party tools). You'll see what we mean if you look at the array of options just in the Colors And Themes section. In the first option there, you can choose a global theme. Breeze is the default KDE Plasma theme, and besides the de rigueur light and dark offerings, it is now available in an in-between Twilight offering, too.



KDE neon Installer

Welcome
Location
Keyboard
Partitions
Users
Install
Finish

About

EFI Select storage device: THNSN5512GPUK NVMe TOSHIBA 512GB - 476.94 GiB (/dev/nvme0n1) GPT

○ **Install alongside**
The installer will shrink a partition to make room for KDE neon Testing Edition.

○ **Replace a partition**
Replaces a partition with KDE neon Testing Edition.

○ **Manual partitioning**
You can create or resize partitions yourself.

Current:

nvme0n1p1 500.00 MiB FAT32	nvme0n1p2 3.00 GiB FAT32	Ubuntu 16.04.7 LTS (16.04) 78.13 GiB ext4	Ubuntu 22.04.4 LTS (22.04) 121.88 GiB ext4
Fedora 32 (Workstation Edition) 40.00 GiB ext4	nvme0n1p7 150.00 GiB ext4	Pop!_OS 22.04 LTS (22.04) 50.00 GiB ext4	nvme0n1p8 16.00 GiB swap

Back | Next | Cancel

The Neon installer coped with our needlessly complicated quadruple boot arrangement with remarkable aplomb.

Go back to the main settings panel and look for Window Management (in the Apps & Windows section). Here you'll find all kinds of knobs to twiddle. If you like eye candy, you'll be powerless to resist the Desktop Effects options. You'll find classic effects (such as Wobbly Windows and the recently resurrected Desktop Cube), as well as accessibility features such as screen magnification and colour blindness corrections. If you're like us (hard of eyesight from years of CRT-induced retina burn) and sometimes lose the cursor on HiDPI displays, the cursor locator effect is a godsend. Once activated, press Windows (referred to as Meta in KDE Plasma) and Ctrl, and a very visible dual-circle arrangement shows you where the cursor is hiding.

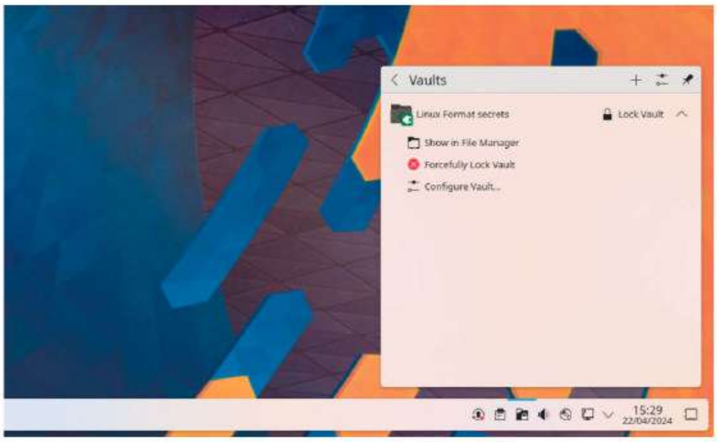
Plasma also gives you a choice of Application Launcher stylings. Right-click it and select the Show Alternatives option. From here you can choose between a classical cascading menu affair, a modern full-screen Dashboard (similar to Gnome but with categories listed on the right), or the default Launcher. All of these are summoned with the Windows/Meta key and all enable you to search just by typing (you don't need to go to the effort of clicking the search bar). You can also directly edit the applications and their order by right-clicking the Launcher icon, then choosing Edit Applications. You can even change the icon for the Launcher button.

»

Getting to know Neon

KDE Plasma and the underlying Neon distro are pretty intuitive, but we've got some pointers to make them even easier.

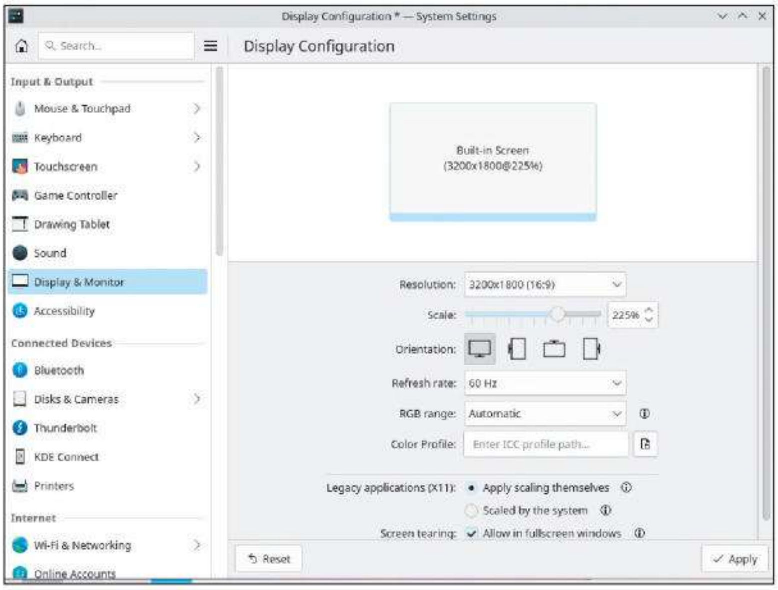
If you (successfully) installed Neon, after rebooting you'll see a monochrome boot menu listing Neon, any other OSes, as well as some memory test options (which you should never need). Choose Neon, and after a few moments you're presented with a login screen. Keen-eyed users will spot the session-choice box in the lower-left and note that it defaults to Plasma Wayland. More on this later. For now, log in with the credentials you gave at installation, and you're greeted with the welcome screen. There's a bit of repetition from the install slideshow here, but if you forgive that, you'll find helpful links to System Settings, KDE's powerful tooling selection and the *Discover* software centre. All of which we'll cover in a moment.



The existence of our super-secret LXF vault was never supposed to be made public. Oops.

The next screen, though, gives you a quite unique slider control over how much telemetry information you share with the KDE team. It's all anonymous, but in theory, if you turned it up to max, your system configuration and usage habits would be unique enough to fingerprint (but not actually identify) you. Then Konqi, who we met earlier, will invite you to contribute to KDE development. And finally Konqi and friend will ask you to consider donating financially to KDE eV, the German non-profit organisation that supports KDE development. Now it's time to start exploring the distro.

Depending on which edition you installed, there will probably be some updates available. And you'll see a notification indicating this (taking the form of an up arrow) in the system tray. Double-click this to open *Discover*. Like Ubuntu's *Software Center* (whose spelling will forever cause umbrage to British English users), *Discover* acts as both an app store (for discovering new software) and a tool for updating currently installed software. If you need to be dealing with documents and spreadsheets (or presentations, diagrams and charts), you might want to install *LibreOffice*. But consider for a second installing the lighter-weight, and built using the same *Qt* toolkit (see over) as Plasma, *Calligra* office suite (or whatever components of it you want). If you need *LibreOffice*, it's easy to find in *Discover*, and if you notice these details, you'll see that the package you download has been styled to fit in perfectly with the Breeze theme.



Built-in Screen (3200x1800@225%)

Resolution: 3200x1800 (16:9)
Scale: 225%
Orientation:
Refresh rate: 60 Hz
RGB range: Automatic
Color Profile: Enter ICC profile path...
Legacy applications (X11): ● Apply scaling themselves
○ Scaled by the system
Screen tearing: ☑ Allow in fullscreen windows

Plasma, as we would hope, supports our QHD screen and even gets the fractional scaling correct.

Since Neon is based on Ubuntu, much of the software (particularly popular applications like *Blender* and *Krita)* that's available only as Snap packages on Ubuntu can be ingested that way on Neon, too. Unlike Ubuntu, though, a default Neon installation includes no Snap packages, so if you are morally opposed to them, you can avoid them altogether. By default, in fact, Snaps are disabled, with Flatpaks (sourced through Flathub) enabled out of the box. You can enable them easily from the settings.

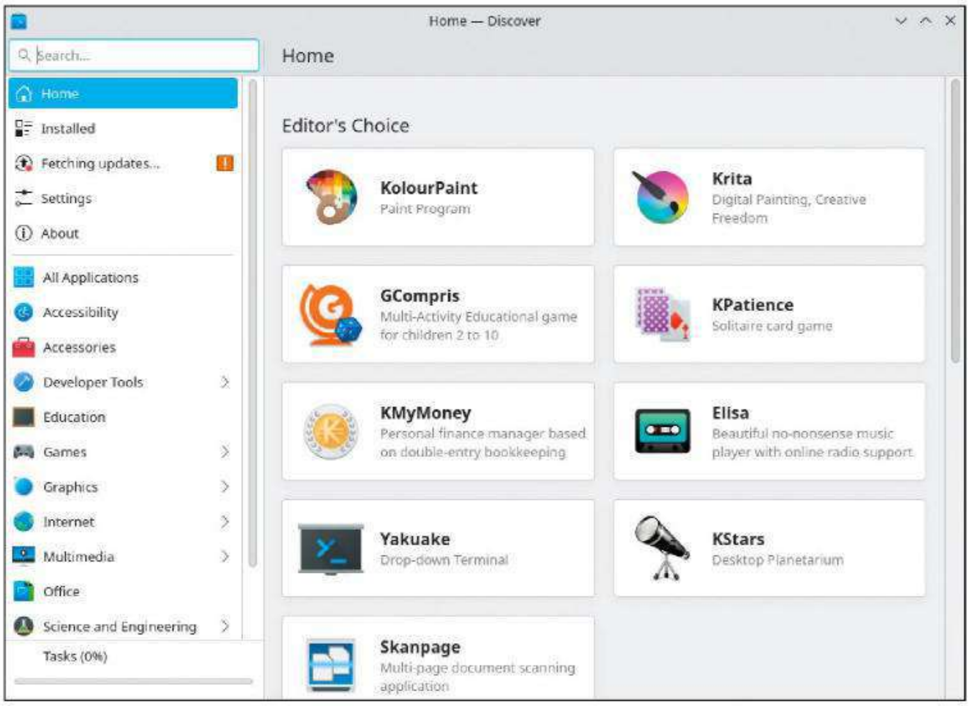
If you use Ubuntu regularly, you might be in the habit of doing updates from the command line with *Apt* – for instance, if something goes wrong with *Software Center* (besides its spelling). If you try to do this with KDE Neon, you'll be gently told off, because its own *Discover* software manager is tightly connected with a different package manager, namely *pkcon* (short for PackageKit config). Thus your `apt update && apt upgrade` habit should be replaced with:

```
$ sudo pkcon refresh
$ sudo pkcon update
```

If you really must, you can still use *Apt*, and it will tell you the alternative syntax that's required. Speaking of Ubuntu, the latest LTS release, 24.04, came out just a little too soon for Plasma 6.0 to be included in the repos (or incorporated as part of Kubuntu). See the box *(below-right)* to see how to use the backports repository to get it.

Get (KDE) Connected

The *KDE Connect* program is popular among users of all desktops, because it's without exception the best tool for dealing with your mobile devices. You can send messages, share files, use a VPN set up on your desktop machine, and much more. It's installed by default on Neon (you'll find it in the Status And Notifications area) and easy to find on other distros. One of its most exciting new features is the ability to use your smartphone as a remote control for your PC, enabling you to shut it down (or perform any number of other automations, depending on your scripting ability) with a single tap.



Discover is your one-stop shop for free software, and we think it's more stylish than both Gnome Software and the Ubuntu Software Center.

To use *KDE Connect,* you need to install the app on whatever iOS or Android (or ChromeOS) device(s) you want it to work with. It's available on the Apple Store, the Play Store, or on F-Droid if you prefer to keep things FOSS. Either way it's free, and once it's installed, you'll want to pair your various devices together. Open the app on two devices on the same network, and you should be able to request a pairing through the app (in either direction). Once you confirm the pairing request, you can (from the PC side) work such magic as see your phone's battery status from the PC, share clipboards between devices or even make your phone ring (very useful if you misplace it). In the other direction, you can use your phone as a remote control (or presentation controller) and with, some plugin installation and permission granting, much more. If you have any issues, check the official documentation at **https://userbase.kde.org/KDEConnect**.

If you crave further integration, you may be interested in the Plasma Integration browser extension for *Firefox*. This connects not only with *KDE Connect* (so you can download files to mobile devices), but also with the rest of the Plasma desktop. It shows download progress and provides media playback controls in the system notification area. It also integrates browser tabs and history in the global search function (if you grant the extra permission in the extension's preferences).

For those with secrets – and, let's face it, we all have secrets (like those three-hour 'off-site meetings' in local ale houses) – then you might be interested in storing them in *KDE Vaults*. A vault acts like a password-protected folder, but is heavily encrypted and has (if you like) a distinct password from the one you use to log in. *Vaults* is also found in the Status area, and when you start it you're invited to create a new vault. You can have multiple vaults and a number of storage back ends/ciphers are available, too (although the default CryFS/AES-gcm combo will be fine for most users). Vaults are stored in the **~/.local/share/plasma-vault** directory with an ENC extension. The CryFS format used here doesn't reveal filenames or other metadata, so it's safe to back the ENC files up to a cloud service to keep them safe. If you're paranoid (or otherwise have reason to believe that you are being surveilled), you can even choose to go offline while a particular vault is open.

» KDE ON OTHER DISTROS

It's entirely possible to install KDE Plasma as an additional desktop (as opposed to installing a whole new KDE-based distro). We slightly discouraged this earlier, because there tend to be some (largely cosmetic) side effects. Namely, you may end up with a whole bunch of extra programs (such as the Gnome text editor and the KDE one), which can be distracting or annoying. Ubuntu offers several KDE Plasma metapackages. If you just want a bare-bones desktop, open a terminal and run:

```
$ sudo apt install kde-plasma-desktop
```

If you want the desktop with Kubuntu's customisations, use `kubuntu-desktop`. Or if you want a fuller KDE experience, go with `kde-full` or `kde-standard`. On Ubuntu 24.04, this will give you Plasma 5.0. To get the latest and greatest, you'll need to enable the Kubuntu backports PPA, which you can do with the following:

```
$ sudo add-apt-repository ppa:kubuntu-ppa/backports
```

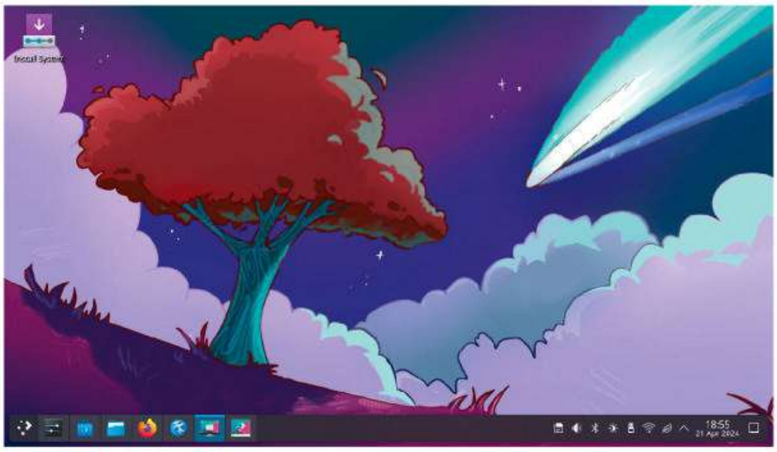
We won't get into it here, but if you're using Lubuntu and want a bit of a change, you can add more KDE style without going full KDE. In particular, you can replace the *Openbox* window manager with *Kwin*, and then install the Breeze theme. That way you'll keep everything *Qt*-based, in case you're a toolkit purist (see over).

Toolkit terminology

Get the low-down on GTK how GIMP fits into this and Qt, and find out about KDE's long road to Wayland, Plasma Mobile and more.

Gnome, KDE Plasma or any other desktop you'd care to name all rely on low-level frameworks to work their magic. The rudiments of any desktop application (buttons, scroll bars, menus, dialogs, tooltips…) are known as widgets and these are a big part of what those underlying frameworks provide. You'll sometimes see *GTK* (the thing that underlies Gnome) and *Qt* (that which underlies KDE Plasma) described as widget toolkits, but this isn't strictly correct as they provide other things, too. You can read more about *GTK* and *Qt* in the box *(below).* Another key component of any desktop environment is the window manager (such as Gnome's *Mutter* or Plasma's *Kwin).* As you'd guess, this manages windows, where they're drawn and how they're decorated.

Unless you've been living under a desktop-Linux-proof rock for the past decade, you'll be aware that Linux desktops are shifting away from the X.org display server and towards the modern Wayland protocol. Without getting into exactly what these things are, they both run at a lower level than the desktop environment and they both put things on the screen (in a sense, that's all Wayland does). The transition to Wayland compatibility relied primarily on making toolkits and window managers work with it. Most Gnome-based distros have defaulted to Wayland for the last couple of years (Fedora 34 introduced the change in 2021, and its latest release does away with the X.org session altogether). And for KDE, the move to Wayland is, if not complete, at least more than ready for everyday use, so it's the default in KDE Neon, too.



Many of the backgrounds, such as the default Scarlet Tree, include a version for dark themes.

As we write this, one of the latest features to be incorporated into *Kwin's* Wayland implementation is Explicit Sync. This allows applications to say with fine precision when to display a particular frame on the screen. The upshot is less latency and fewer graphical glitches (tearing or other corruption), particularly on Nvidia hardware. For a long time, Nvidia's proprietary driver did not work well with Wayland, but mercifully those days are behind us. So, the Linux desktop moves ineluctably towards Wayland, and as it does so the comparatively few developers who still understand X.org will inevitably reduce their maintenance of it.

Be that as it may, if you have problems with the Wayland session (such as a blank screen after logging in), you can force an X.org session by clicking in the lower-left where it says Desktop Session and selecting Plasma (X11). If you don't get that far, and get no login screen at all, you may need to tell the login manager (SDDM) itself to not use Wayland. This indicates a serious bug or hardware issue, which you should tell the developers about, but you can work around it by

EARLY DAYS

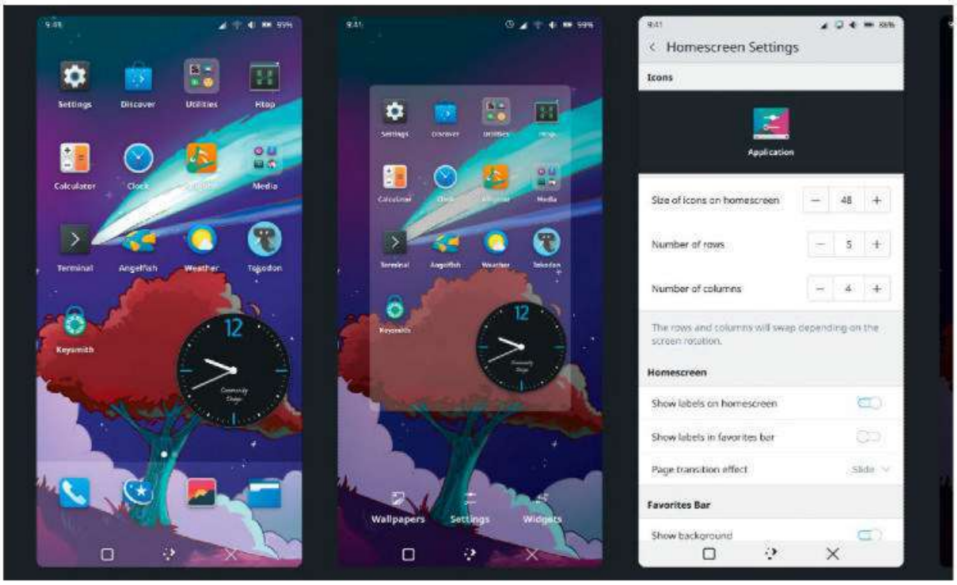
GTK and *Qt* have very different origin stories, which we'll try to synopsise. The former was built for the venerable *GIMP (Gnu Image Manipulation Program),* when a developer become tired of the *Motif* toolkit. It saw its first outing in 1999 and was naturally called the *GIMP Toolkit (GTK).* It included all the funky paintbrush and bucket iconography used in *GIMP's* toolbox dialog. *Qt* was originally a dual-licensed product released by Trolltech (who designed it for an ultrasound imaging program it was working on for X11 and Windows) in 1995, and used by KDE since its inception in 1996. Interestingly, the first enterprise customer of *Qt* was the European Space Agency.

Controversy around the licence (the 'free' side of which forbade redistribution, flouting one of Stallman's Four Freedoms) was resolved, to some extent, in 1999 with the introduction of the Q Public License (QPL, which only applied to the X11 edition) and the promise that it would default to a BSD-style licence in the event of no release happening. The issue was finally settled the next year when *Qt 2.2* was released under GPL v2. Trolltech was acquired by Nokia in 2008, who (then ruler of the mobile sphere) wanted to use *Qt* in its phones. That didn't work out so well, and Nokia switched to Windows mobile (which went even less well). Commercial *Qt* operations were sold to Digia, and today they're run by its erstwhile subsidiary The Qt Company.

Open source development of *Qt* today is governed by the Qt Project, while *GTK* development has been (mostly) under the aegis of the Gnome Foundation since its inception in 2000.

editing the SDDM configuration file at **/etc/sddm.conf** and adding:

```
[General]
DisplayServer=x11
```



Homescreen Settings

Icons

Application

Size of icons on homescreen − 48 +

Number of rows − 5 +

Number of columns − 4 +

The rows and columns will swap depending on the screen rotation.

Homescreen

Show labels on homescreen

Show labels in favorites bar

Page transition effect Slide

Favorites Bar

Show background

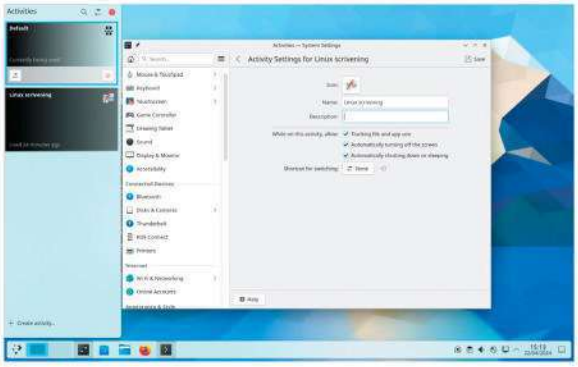
Plasma 6 Mobile, coming to a Linux-powered phone near you soon.

Theme park

GTK 2.0, and to some extent *3.0,* allowed quite extensive theming. And in *Qt* it has more or less always been a thing. *GTK 4.0* took the bold step of having only one theme, and rendering everything through the libadwiata library. The reason for *GTK's* slightly austere ruling here is that allowing theming in *GTK 3.0* made some apps look, and in many cases behave, bad. This upset theme makers and in particular Pop!_OS devs *(see our* **LXF290** *Top Of The Pop!_OS feature where System 76 Rust boffin Michael Murphy discusses this),* whose stylish looks were achieved in a large part through extensive theming. System 76 is now working on a brand-new desktop powered by Rust, which we can't wait to try.

Theming in KDE Plasma is a multi-level affair. We've already mentioned the Breeze Light, Dark and Twilight species. These are known as Global Themes, as they cover not just how everything looks (buttons, fonts, borders, window controls, accent colours) but also how everything acts. Beneath the Global Themes are controls for the Application Style (you'll find it in the Colours & Themes settings), which controls how widgets inside programs look. If you're feeling retro (or enjoy chunky tabs, right angles and heavy shadows), you might want to apply the MS Windows9x Style. There's also a Fusion style, which falls between Breeze's own app style and the former. You can further customise the active style by clicking the Edit button.

There are more style classes, in particular the Plasma style, which governs how the desktop itself looks, including things like panel layout, colour scheme and desktop widgets. User-contributed Plasma styles and Global Themes are available from the Get New button. But do heed the warning before downloading anything. These are all contributed by third parties and, beyond community review, aren't vetted for quality or security. In March 2024, some ne'er do well uploaded a Global Theme that included the unwanted side effect of deleting all the files any user who installed it had access to. Unfortunately, this type of behaviour will always be possible, since Global Themes allow you to rewrite the behaviour of the whole desktop. So it's trivial to bury a cheeky `rm -rf / --no-preserve-root` in a script. If you stick to popular themes, you'll be fine. You can also theme the login and splash screens.



Activity Settings for Linux scrivening

Name: Linux scrivening

Description:

While on this activity, allow: ✓ Tracking file and app use ✓ Automatically turning off the screen ✓ Automatically shutting down or sleeping

Shortcut for switching: None

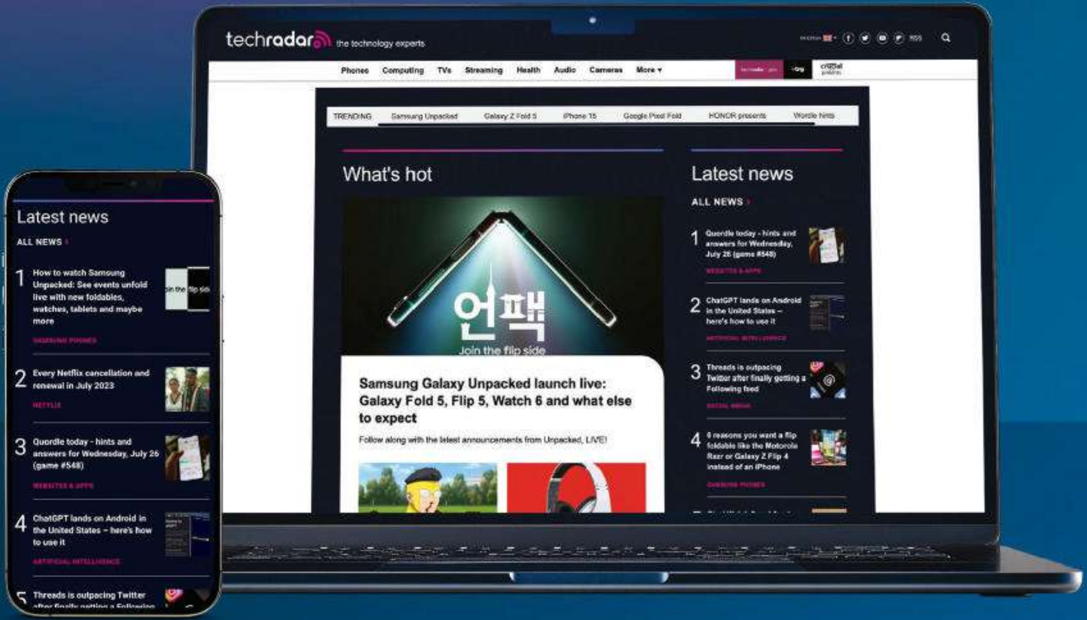
We devoted an entire activity to our writing work, but spent more time customising it than actually working.

Pressing Meta and W (or doing a four-finger pinch on a touchscreen) brings up the desktop Overview mode. Here you can see all the virtual desktops, or create a new one. Note that the speed and effects of all the transitions between views and virtual desktops can be configured. Above virtual desktops, Plasma enables you to define Activities, so you can have several virtual desktops housing all your programs related to one task, and all other programs and desktops invisible. You can define a new Activity from the Settings panel in the Apps & Windows section. You might, for example, want separate Activities for gaming and working. You can then switch Activities by pressing Meta and A. Slightly awkwardly, you can also get an Activities Overview by holding down Alt, then pressing D and then A. Activities can be combined with multiple monitors, and apps can be forced to open in particular Activities, so that your workflow remains pure.

We've got this far and somehow managed to ignore the fact that KDE Plasma is used in the Steam Deck, one of the most popular portable gaming devices in existence. Obviously, not very many Steam Deck users (Deck-heads?) use it to do desktop things, beyond installing the likes of *ProtonUp-Qt* and *Heroic Games Launcher.* But hopefully this outing encourages them to delve deeper. The Arch Linux-based SteamOS still uses Plasma 5, but will surely upgrade in time.

In a fine demonstration of journalistic splendour, we've also not mentioned Plasma Mobile. As you might guess, it's a user interface for mobile devices and is available for Pine64 devices and the Librem 5. Thanks to big parts of Plasma being device independent and the Kirigami framework (already used by many desktop Plasma programs), it's easy to make apps that are truly convergent. We look forward to covering all things Plasma Mobile, including how to get it running with PostMarketOS, in the future. Oh, and we'll surely have more KDE news and features soon, too! LXF

Meet the technology experts



techradar the technology experts

Phones Computing TVs Streaming Health Audio Cameras More ▾

TRENDING Samsung Unpacked Galaxy Z Fold 5 iPhone 15 Google Pixel Fold HONOR presents Wordle hints

What's hot

Join the flip side

Samsung Galaxy Unpacked launch live: Galaxy Fold 5, Flip 5, Watch 6 and what else to expect

Follow along with the latest announcements from Unpacked, LIVE!

Latest news

ALL NEWS ›

1 Quordle today - hints and answers for Wednesday, July 26 (game #548)
WEBSITES & APPS

2 ChatGPT lands on Android in the United States – here's how to use it
ARTIFICIAL INTELLIGENCE

3 Threads is outpacing Twitter after finally getting a Following feed
SOCIAL MEDIA

4 6 reasons you want a flip foldable like the Motorola Razr or Galaxy Z Flip 4 instead of an iPhone
SAMSUNG PHONES

Latest news

ALL NEWS ›

1 How to watch Samsung Unpacked: See events unfold live with new foldables, watches, tablets and maybe more
SAMSUNG PHONES

2 Every Netflix cancellation and renewal in July 2023
NETFLIX

3 Quordle today - hints and answers for Wednesday, July 26 (game #548)
WEBSITES & APPS

4 ChatGPT lands on Android in the United States – here's how to use it
ARTIFICIAL INTELLIGENCE

5 Threads is outpacing Twitter

- The world's **most** comprehensive technology website
- An **unrivalled** mix of news, opinions, reviews and features
- **All-new design**, new homepage, new features and special reports
- Backed by **over 300 years** of editorial experience

techradar
the technology experts

www.techradar.com

Pi Foundation co-founder Jack Lang dies aged 76

Lang was a "central part of the Cambridge tech entrepreneurship scene" and helped establish the Pi.

Emmanuel College, Cambridge, posted the sad news that Jack Lang, one of the original co-founders of the Raspberry Pi Foundation, passed away on 23rd April after a long illness. Born in 1948, Lang matriculated into Emmanuel College in 1966. He was a well-known entrepreneur and business angel in Cambridge. For Raspberry Pi fans, he is perhaps best known as one of the original co-founders of the Raspberry Pi Foundation, the educational charity linked to the popular single-board computer.

Lang was also a trustee and chairman of the board of the Raspberry Pi Foundation. He also became a non-executive director and chairman of the board of Raspberry Pi Ltd, the Raspberry Pi trading arm, which oversees development of the single-board computers.

Raspberry Pi CEO Eben Upton added his memories: "It's very sad news. Jack was an old friend, and had been a central part of the Cambridge tech entrepreneurship scene, having played an instrumental role in the development of the software for the BBC Micro, since long before I arrived in 1996."

A fun part of Pi history saw Lang store some of the original batch of Pi boards in his garage before a select few were sent off for EM testing and certification. The Emmanuel College news post sums Lang up rather well: "Jack was one of the most significant figures in computing education in the UK." He'll be sadly missed. Discover more about his apple growing, sourdough making, Michelin star winning and firework making antics: **www.emma.cam.ac.uk/about/news/?id=683**

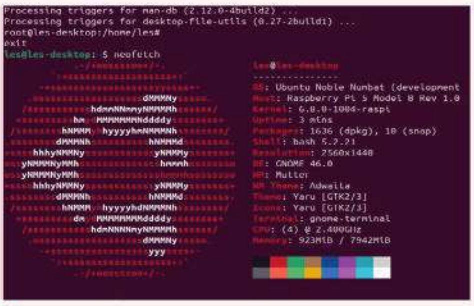
Pi Foundation co-founder Jack Lang.

CREDIT: Raspberry Pi Foundation

Booting Ubuntu

...or not.

It seems installation of the Pi version of Ubuntu 24.04 isn't going as smoothly as hoped. A combination of USB drives, SD card and NVMe can cause install issues on both the Raspberry Pi 4 and 5 models. Investigations are ongoing and workarounds in place. Get all the details: **https://bit.ly/lxf316bug**



```
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for desktop-file-utils (0.27-2build1) ...
root@les-desktop:/home/les#
exit
les@les-desktop:~$ neofetch
                                              les@les-desktop
                                              ---------------
                                              OS: Ubuntu Noble Numbat (development
                                              Host: Raspberry Pi 5 Model B Rev 1.0
                                              Kernel: 6.8.0-1004-raspi
                                              Uptime: 3 mins
                                              Packages: 1636 (dpkg), 10 (snap)
                                              Shell: bash 5.2.21
                                              Resolution: 2560x1440
                                              DE: GNOME 46.0
                                              WM: Mutter
                                              WM Theme: Adwaita
                                              Theme: Yaru [GTK2/3]
                                              Icons: Yaru [GTK2/3]
                                              Terminal: gnome-terminal
                                              CPU: (4) @ 2.400GHz
                                              Memory: 923MiB / 7942MiB
```

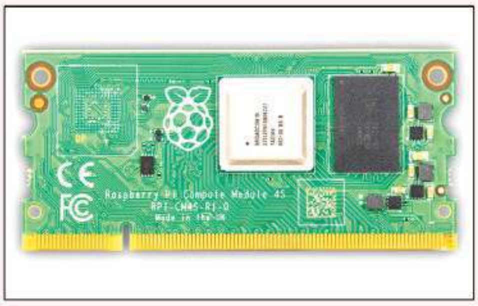
Not so noble after all, it might seem...

CREDIT: Raspberry Pi Foundation

Moah memory!

Compute that baby.

Since 2014, the Compute Module has offered industrial solutions in a flexible form factor. To keep with the times, the 4S model is now available in 1GB, 2GB, 4GB and 8GB options. It's a drop-in replacement for the old Module 3 version but based on the Pi 4, so has significantly more oomph, as the boffins like to say. Read more: **https://bit.ly/lxf316mem**



We can't see the extra memory from here.

Les Pounder works with groups such as the Raspberry Pi Foundation to help boost people's maker skills.

» MAKE Pi NOT WAR

I've recently come back from Pi Wars (**https://piwars.org**), a Raspberry Pi robotics event hosted at the William Gates Building, Cambridge (the same place that I first interviewed Eben Upton back in 2013). Pi Wars was created by Mike Horne and Tim Richardson as a non-destructive version of Robot Wars. Teams from around the world (USA, Netherlands and Nigeria) took part in a series of challenging courses and tasks.

Day one saw younger roboteers take part and it was awesome to see so many eager children learning to code and build robots using the Raspberry Pi. From simple, cheap robots, all the way to incredibly sophisticated builds.

On the second day, it was the turn of the adults, and their builds were more complex than the younger makers, but both age groups had an exceptional level of robots.

When the Raspberry Pi first came out, it was hard to understand how makers would use the low-cost single-board computer, but in the 12 years since, we now see a plethora of awesome projects. All of which have been made possible thanks to the Raspberry Pi.

The Raspberry Pi 5, released in late 2023, is an absolute monster when compared to the original Raspberry Pi, making AI and computer vision projects much easier, thanks to its considerable horsepower. If you don't need the power, but still want a great platform, the Raspberry Pi Pico W is a great place to start building robots. For £6, you get a capable board, and robot kits can be bought for less than £20.

Dosbian 2.5

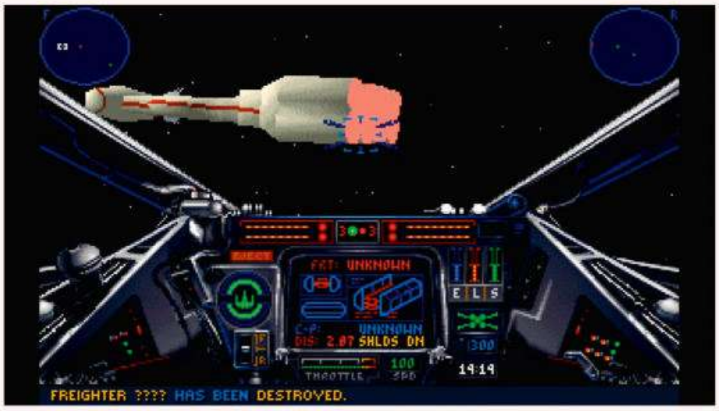
Les Pounder lives in the past – he loves old computers and can remember a time when Snickers were Marathons.

IN BRIEF

An x86 PC emulator for the Raspberry Pi 3, 4, 400 and now 5. Essentially the *DOSBox* emulator on top of Debian Bookworm, the OS auto-starts in a DOS environment, where we can quickly set up and play our games, and emulate old operating systems. It is a little tricky to get started, but once you understand, it all works beautifully.

MS-DOS is where this writer cut his teeth with PCs. Before that it was the Commodore 64 and Amiga. He felt at home at the command prompt, and it explains why he loves Linux so much. What if we want to relive the heady days of DOS prompts, IRQ configuration and batch scripts?

Dosbian 2.5 is the latest version of the popular DOS emulation distro and the first to support the Raspberry Pi 5. We can run Dosbian on any model of Raspberry Pi 3, 4, 400 and 5. Under the hood we are running Debian Bookworm, the same as the official Raspberry Pi OS. On top of that is a version of DOS that we can use to run our old games and apps. We're not limited to just DOS; we can comfortably emulate up to Windows 98.

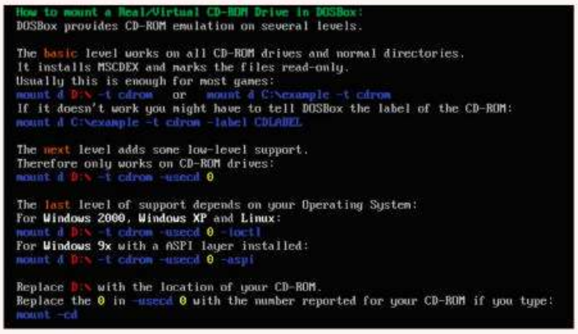


In the '90s, this was your best chance to fight against the Empire, and with Dosbian, we get that chance again in 2024.

Dosbian has three versions of the x86 *DOSBox* emulator. The default is *DOSBox*, but there is an alternative in *DOSBoxece,* which is older but runs DOS games better. Lastly there is *DOSBox-staging,* the most modern version, which runs most games. For those of us that grew up on Lucasfilm Games and point-and-click adventures, *ScummVM* is included, *Scumm* (Script Creation Utility for Maniac Mansion) being the system used to create those games.

Flashing Dosbian to a microSD card, we booted up our Raspberry Pi 5 and were instantly greeted by the DOS prompt. Ah, good memories! From here we can try out the included shareware games, found in the **Games** directory. We spent a while 'testing' *Doom* to ensure that it worked correctly. All was good.

How can we get our own games in Dosbian? Luckily, we have a Linux OS under the hood. Mounting a USB drive full of games was easy and from there we used the *mount* command inside of Dosbian to match the USB device under Linux to a drive letter for DOS. We also had *Star Wars X-Wing* on CD-ROM, a game we poured many hours into as teenagers. Using *imgmount*, we mounted the ISO as a drive, ran the installer and were soon fighting Imperials – after we remembered how to control the X-Wing. Initially we forgot to run the installer, running the game without sound. The installer checked our virtual soundcard, configured the IRQ/DMA and so on, and we were ready to fully experience space dogfights!



```
How to mount a Real/Virtual CD-ROM Drive in DOSBox:
DOSBox provides CD-ROM emulation on several levels.

The basic level works on all CD-ROM drives and normal directories.
It installs MSCDEX and marks the files read-only.
Usually this is enough for most games:
mount d D:\ -t cdrom   or   mount d C:\example -t cdrom
If it doesn't work you might have to tell DOSBox the label of the CD-ROM:
mount d C:\example -t cdrom -label CDLABEL

The next level adds some low-level support.
Therefore only works on CD-ROM drives:
mount d D:\ -t cdrom -usecd 0

The last level of support depends on your Operating System:
For Windows 2000, Windows XP and Linux:
mount d D:\ -t cdrom -usecd 0 -ioctl
For Windows 9x with a ASPI layer installed:
mount d D:\ -t cdrom -usecd 0 -aspi

Replace D:\ with the location of your CD-ROM.
Replace the 0 in -usecd 0 with the number reported for your CD-ROM if you type:
mount -cd
```

Dosbian is essentially DOSBox on top of Raspberry Pi OS. We get multiple versions of DOSBox, each with its own strengths.

As well as a 'pure' DOS interface, we also have *LaunchBox*, a front-end aimed at gaming. From there we can run our favourite games; we just need to put a little work in to creating menu items, genres and so on. The rather retro user interface is simple to use: we name our game, pass the command to run, and any additional configuration commands. We can also delve into Details and from there set up our game genre, publisher, year and so on. Unlike other emulation front-ends, there is no scraping tool to obtain the information.

Dosbian is great fun, but let's be honest, at first it can be a little tricky. How we get our games into Dosbian being the most pertinent. The absolute easiest way is to power down the Pi, remove the SD card and insert it into a Linux PC. Copy your legally (*good boy!–Ed*) owned games over and then return the card to the Pi. A quick power-up and your games are in their respective folders.

You're going to have a great time with Dosbian – the learning curve shouldn't put you off. Persevere and enjoy PC gaming and computing from the '80s and '90s! LXF

VERDICT

DEVELOPER: Carmelo Maiolino
WEB: https://cmaiolino.wordpress.com/dosbian/
LICENCE: Donationware

FEATURES	9/10	EASE OF USE	7/10
PERFORMANCE	9/10	DOCUMENTATION	6/10

Solid DOS performance allows us to enjoy classics without too much config. A great way to get into retro PC computing.

» **Rating 8/10**

Creality K1C

Constantly demanding perfection, **Denise Bertacchi** is finally pleased with Creality. Well, almost.

SPECS

Build: 220x220 x250mm
Type: PLA/ PETG/TPU/ABS (up to 300°C)
Extruder: Direct drive
Nozzle: 0.4mm (Unicorn)
Bed: Coated steel flex plate, heated
Levelling: Automatic
Sensor: Runout
Comms: LAN, Creality Cloud, USB flash disk
Control: 4.3-inch colour touchscreen
Size: 355x355 x482mm, 12.4kg

The K1C feels like a placeholder from a company that wants to constantly impress customers with shiny new products. Clearly, Creality wanted to show off its improvements to the K1's design, but wasn't ready to announce the K2 – a name we now know is being reserved for a four-colour 3D printer.

The C in K1C stands for Carbon, which is a bit of a red herring. You don't need to redesign an entire printer to tackle abrasive carbon filled materials; all you need is a durable nozzle, which can be purchased for a few pounds on Amazon. Don't get us wrong, the K1C is a superior printer, and the new steel-tipped Unicorn nozzle is definitely an improvement, but all this could have waited until the colour module was ready for retail.

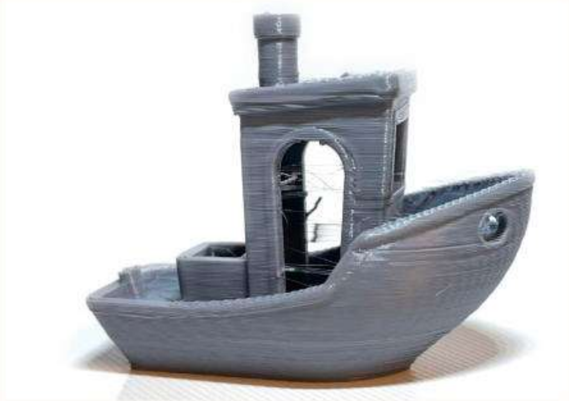
One of the other improvements to the K1C is an all-metal extruder, an AI-assisted camera that comes as standard, a cabin filter and an improved levelling system. It remains the same size and speed as the K1, making it one of the best 3D printers you can buy.

The Creality K1C comes with everything you need to get your printer set up. You get tools to maintain the printer, side cutters, a plastic scraper and a USB stick. There's also a sample spool of white Creality Hyper PLA to start your printing journey. You get a PDF copy of the manual, a copy of *Creality Slicer* and sample models in both pre-sliced GCODE and STL format, too. There's also a very helpful video to help you operate your machine.

Klippered wings

The build plate is a smooth PEI-coated sheet that requires glue stick – annoying, but it works. It does not have Lidar, which is still reserved for the K1Max. It ships with a limited version of Klipper modded by Creality, but those interested in going full vanilla Klipper can easily root the firmware, as Creality has left the door open.

The biggest change on the K1C is the tri-metal nozzle, which Creality has dubbed a Unicorn style. This is a long, one-piece nozzle made from copper, with a titanium alloy heat break and a steel tip. The nozzle is proprietary, but they're easily available at Creality's store and on Amazon.



Considering the speed at which this was done, it's an excellent Benchy result.

There's a 3D printer in there somewhere and boy it's fast!

Clearly, Creality has been listening to user feedback, as it has provided a printable solution to the annoying rear-mounted spool holder. Two printable mounts are included on the printer's USB stick, with a bag of screws and directions on how to attach it. Like the K1, this machine has excellent cooling and an enclosure that makes printing ABS and ASA very easy.

The Creality K1C is 98% assembled and only needs to be unboxed and unpacked. There's a full auto-levelling program that includes input shaping with the help of built-in sensors. Loading filament is a little awkward with the back-mounted spool holder. The side-mounted spool holder makes it easier, but you must print it first. Creality includes a copy of *Creality Print,* a custom program that integrates file retrieval from the Creality Cloud, with slicer and file delivery. Using the Cloud is optional.

We ran Benchy using Speed Benchy rules: 0.25mm layer height, two walls, three top and bottom layers, 10% infill. We also turned off combing and z hop, and let her rip using the default speed of 300mm/s. The Benchy is a bit stringy, but the layers are fairly even, the curves are nice and there's no ringing. This Benchy printed in an eye-popping 13 minutes and 53 seconds, currently the fastest Speed Benchy we've recorded. LXF

VERDICT

DEVELOPER: Creality
WEB: http://crealityofficial.co.uk
PRICE: £539

FEATURES	9/10	EASE OF USE	9/10
PERFORMANCE	9/10	VALUE	9/10

Offers what we expected from the original in 2023. Fixes issues with tool head and makes AI camera a standard feature.

» **Rating 9/10**

PI PICO

Build your own weather machine

Les Pounder is normally a ray of sunshine, but is feeling a little under the weather. Perhaps this project will lift the clouds.

OUR EXPERT

Les Pounder is associate editor at Tom's Hardware and a freelance maker for hire. He blogs about his projects at **http://bigl.es.**

YOU NEED

- **Pi Pico W**
- **I2C LCD**
- **4x F2F jumper wires**
- **Code: https://github.com/lesp/LXF316/archive/refs/heads/main.zip**

The Raspberry Pi Pico W excels at being used with online APIs for a myriad of data sources. We've used it to read the news, tell us who is on the International Space Station, and now we are building a portable weather station that uses OpenWeather's API to tell us if we need an umbrella.

W is for weather

Holding the BOOTSEL button, connect your Pi Pico to your computer. Go to **https://bit.ly/LXF306MicroPython** and download the version of MicroPython for your Pico/Pico W. Open your file manager, go to the downloaded file and copy it to the root of the **RPI-RP2** drive. This flashes the new firmware to the Pico.

Using your distro's package manager, install *Thonny*. For the latest Ubuntu release, we have to use a Snap:

```
$ sudo snap install thonny
```

Open *Thonny* and connect the Pico to your machine. Go to Tools > Options > Interpreter tab. Set the interpreter to MicroPython (Raspberry Pi Pico) and Port to match the Pico's location. Click OK. *Thonny* connects to the board and we can start writing code.

First we need to install libraries (modules) to enable the Pico to talk to the I2C LCD screen. You can use *Pip/Mip* or *Pypi* to install, but we encountered issues, so we chose to install by recreating the files on the Pico.

We're using dhylands' Python_lcd project (**https://github.com/dhylands/python_lcd**). Create a new blank file in *Thonny*, then open a browser to **https://bit.ly/LXF316dhylands** and copy the contents to *Thonny*. Save the file to the Raspberry Pi Pico as **lcd_api.py**.

Create another blank file, go to **https://bit.ly/LXF316pico_i2c_lcd** and copy the contents to *Thonny*. Save the file as **pico_i2c_lcd.py** to the Pi Pico.

Create another blank file to contain any passwords or API keys we need. In the file create three objects for your Wi-Fi SSID, password and your OpenWeather API key. Replace the text inside the quotation marks with your Wi-Fi details, do not delete the quotation marks as this tells Python that the data inside are strings:

```
SSID = "WIFI SSID"
PW = "WIFI PASSWORD"
openweather = "OPENWEATHER API"
```

How can we get an API key for OpenWeather? Take a look at the boxout *(right)* for more information.

This simple build shows how few components we need to build a weather station using an online API.

Save the file to the Raspberry Pi Pico as **secrets.py** and now we can move on to the main section of code.

Create a new blank file on the Pico and import a series of modules. Network is used to make Wi-Fi connections, urequests is then used to fetch data over that network connection. The machine module has two classes, **I2C** and **Pin**, that we use to make a connection to the I2C LCD screen. Time's **sleep** function enables us to pause the code between weather updates. Next we import dhylands' LCD module to send information to the LCD screen. The last import is our **secrets** file, which contains the Wi-Fi and API details:

```
import network
import urequests
from machine import I2C, Pin
from time import sleep
from pico_i2c_lcd import I2cLcd
from secrets import SSID, PW, openweather
```

The next section of code handles setting up our LCD screen. We need to tell the Pico where it is on the I2C bus. We're using the I2C channel 0, and we need to find the device on the bus. Each device has a unique ID, so we'll scan the bus and get the address, then use it to connect. Then we tell the LCD module that we have a screen with two lines that is 16 characters wide. Lastly, we turn on a blinking cursor to confirm the connection has been made:

```
i2c = I2C(0, sda=Pin(0), scl=Pin(1), freq=400000)
I2C_ADDR = i2c.scan()[0]
```

```
lcd = I2cLcd(i2c, I2C_ADDR, 2, 16)
lcd.blink_cursor_on()
```

To connect to the internet we need to enable the Wi-Fi chip on the Pico:

```
wlan = network.WLAN(network.STA_IF)
wlan.active(True)
```

The next section is a conditional test to keep trying to make an internet connection. If the connection is not made (False), the code tries to reconnect, prints 'No connection' to the REPL and the LCD screen, pauses for two seconds, clears the screen and tries again.

```
while wlan.isconnected() == False:
  wlan.connect(SSID, PW)
  print("No connection")
  lcd.putstr("No connection")
  sleep(2)
  lcd.clear()
```

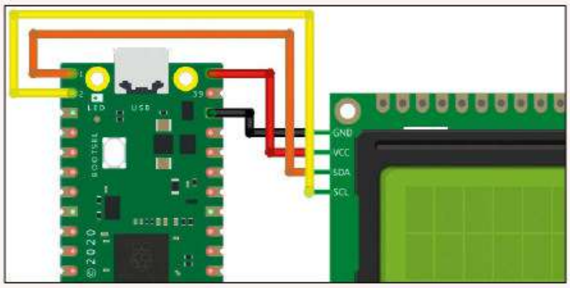
The **else** part activates when the connection is made. The IP address is stored into an object, **ifconfig**, and then printed to the REPL and the LCD screen for five seconds before the display is cleared.

```
else:
  ifconfig = wlan.ifconfig()
  lcd.backlight_on()
  print(ifconfig[0])
  lcd.putstr("IP ADDRESS")
  lcd.putstr("\n"+ifconfig[0])
  sleep(5)
  lcd.clear()
```

Now we get the weather details using the API and display them on the LCD screen. We start with a **while True** loop, then a **for** loop, which flashes the backlight of the LCD screen three times and prints a message to the user to say that the data is being fetched.

```
while True:
  for i in range(3):
    lcd.backlight_off()
    lcd.putstr("Fetching\nweather")
    sleep(0.5)
    lcd.backlight_on()
    sleep(0.5)
    lcd.clear()
```

Out of the **for** loop but still inside **while True**, we create an object, **r**, and use that to store the fetched data from the API. We are using a URL request to fetch the data from OpenWeather. We need to change the location (unless you want Blackpool's weather) by altering **q=Blackpool, UK** to your own location. The **+openweather+** section of the URL inserts our API key.



The circuit uses an I2C LCD screen, connected to the Pico using just four wires.

```
  r = urequests.get("http://api.openweathermap.org/
data/2.5/weather?q=Blackpool,UK&appid="+openweat
her+"&units=metric").json()
```

Create two objects, **weather** and **temperature**, and extract general weather data and temp from the API:

```
  weather = r["weather"]
  temperature = r["main"]["temp"]
```

Update **weather** to contain the **weather** condition:

```
  weather = (weather[0]["main"])
```

Clear the LCD screen and ensure the backlight is on:

```
  lcd.clear()
  lcd.backlight_on()
```

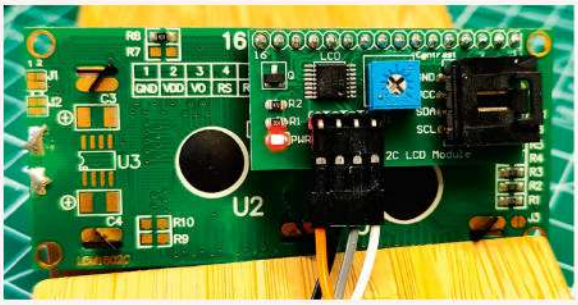
On the top line of the LCD screen, write the current weather condition. On the second line, write the current temperature. We use the **\n** escape character to move to a new line, then we write Temp: to the display. Using concatenation we join the contents of the temperature object to the string. An additional concatenation adds C to identify degrees Celsius. The code pauses for five minutes (300 seconds) before clearing the screen and repeating the **while True** loop.

```
  lcd.putstr(weather)
  lcd.putstr("\nTemp: "+str(temperature)+"C")
  sleep(300)
  lcd.clear()
```

Save the code as **main.py** to the Pico. This forces it to run when the Pico W receives power. To test the code, click on Run > Run Current Script, or click on the green play button. You should see your Pico W connect to the Wi-Fi and retrieve your location's latest weather.

Building the circuit

The circuit is simply four connections between the Pico and the LCD screen *(see diagram above)*: power, GND and two I2C (inter-integrated circuit) connections – SDA (data) and SCL (clock). We're using an I2C screen rather than a traditional HD44780 because it reduces the wiring from over a dozen to just four.



LCD screens normally need over a dozen wires; with an I2C backpack, this is reduced to just four.

» OPENWEATHER API

OpenWeather is a great, free resource for current, historical and predicted weather. To use it, you need an API key. There is no cost, just sign up and be respectful of the service. Go to **https://home.openweathermap.org/users/sign_in** and create an account. Log in and click on API Keys. Under Create Key, name the key and click Generate. The key appears on the left. Copy and paste it inside the quotation marks for the **openweather** object in your **secrets** file. Don't share or post the API key online – it is linked to your name.

» GET YOUR Pi FILLING HERE Subscribe now at http://bit.ly/LinuxFormat

Inside Linux

As we delve into what makes Linux tick, your guide **Matt Holder** explains how abstraction keeps you safely away from hardware voltages.

CREDIT: Magictorch

In this new series of articles, we are going to investigate some of the plumbing that is used to give us the incredible experience that we are used to with Linux distros. This ranges all the way from the kernel that performs interactions directly with the hardware to Wayland, the display server used to support our graphical user interfaces. Along the way, we will also meet PipeWire, which is used for audio and video routing around our systems. In some cases, we will also discuss the tools that these technologies are on the way to replacing, such as PulseAudio for audio playback and the venerable X in the case of Wayland.

Let's begin at the beginning, introducing the Linux kernel, as this performs the direct interactions with your hardware and then provides methods for our software to interact with it.

In 1991, a Finnish student began work on a hobby project, which he called Linux. This was the humble beginnings of what we now use to interact with the hardware in our computers. Most people refer to Ubuntu or Fedora as a Linux distribution. Technically this is incorrect, as a lot of other tools are required to complete the operating system. One school of thought is that we should refer to these operating systems as GNU/Linux, which references the large number of GNU tools that are included.

Kernel architecture

Ubuntu uses the Linux kernel. By design this is a monolithic kernel, which means it carries out a lot of functions all running in kernel space (running with the same privileges). A monolithic kernel handles inter-process communication (IPC), filesystems, scheduling, virtual memory and device drivers, to name a few. In contrast, the footprint of a microkernel is as small as possible. In terms of functionality, a microkernel manages IPC, scheduling and virtual memory. Everything else is managed in userspace by the kernel being able to communicate with corresponding services/daemons. For example, a device driver can communicate with the hardware by using a service that provides any data processing needed before the kernel then communicates this with the hardware.

Which kernel design has seized the day? Well, looking at examples of monolithic kernels – Linux, BSD, DOS, Unix – it seems that monolithic is the most popular. One school of thought is that having a kernel performing as few tasks as possible allows the code to be more rigorously tested and be more stable than a monolithic equivalent. Maybe a device driver crashes – using the microkernel example, the OS itself would not crash, but this hardware would be unavailable. A third kernel design exists, hybrid, which is a mixture of the two. The Windows NT kernel follows a hybrid model.

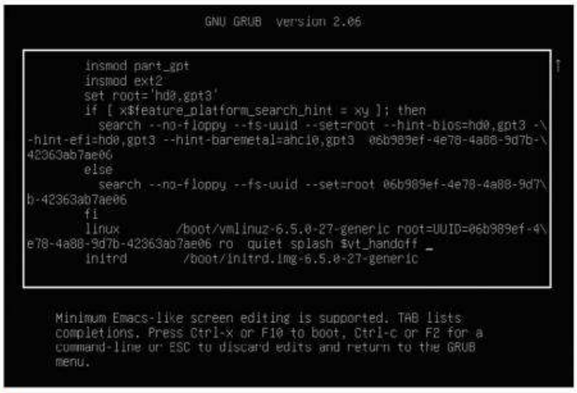
Versions, versions everywhere...

The Linux kernel consists of multiple versions, which are all supported for different amounts of time. Every nine to ten weeks, a new kernel is released and this is called mainline. Next we have the stable release, which contains backports for bug fixes from mainline. The long-term release is supported for a number of years and can be used where complete stability is required. Finally we have the linux-next release, which is where new functionality is tested. Distros often offer their own long-term kernel releases, which may or may not be based on the work completed by kernel developers.

Subsystems and layers:

The kernel can be viewed as being separated into functions and layers. Each of the six functions is split into six layers. On the left-hand side of the diagram *(see next page)*, we can see the name of the layers. We will start from the bottom-left corner to the top-right. At the lowest level of the stack are the electronics of the hardware itself, moving all the way up to interfaces used by userspace. Note: not all subsystems require the bridges layer and some subsystems are interconnected, even between different layers.

Let's take the example of the memory column. The top layer is memory access, which programs use to interact with the physical memory. Next we have the virtual memory layer, which can use both hard disk-based virtual memory and the memory in the device itself. The next layer down is that of memory mapping. Here we see a link to virtual filesystems (which we will look at in a later article) and the swap system (which itself links to the block devices layer of the storage subsystem). At the next level we have the logical memory, followed by the page allocator, which also interacts with the storage subsystem. Finally, the page allocator layer interacts directly with the hardware.



```
                    GNU GRUB  version 2.06

        insmod part_gpt
        insmod ext2
        set root='hd0,gpt3'
        if [ x$feature_platform_search_hint = xy ]; then
          search --no-floppy --fs-uuid --set=root --hint-bios=hd0,gpt3 -\
-hint-efi=hd0,gpt3 --hint-baremetal=ahci0,gpt3  06b989ef-4e78-4a88-9d7b-\
42363ab7ae06
        else
          search --no-floppy --fs-uuid --set=root 06b989ef-4e78-4a88-9d7\
b-42363ab7ae06
        fi
        linux       /boot/vmlinuz-6.5.0-27-generic root=UUID=06b989ef-4\
e78-4a88-9d7b-42363ab7ae06 ro  quiet splash $vt_handoff _
        initrd      /boot/initrd.img-6.5.0-27-generic

    Minimum Emacs-like screen editing is supported. TAB lists
    completions. Press Ctrl-x or F10 to boot, Ctrl-c or F2 for a
    command-line or ESC to discard edits and return to the GRUB
    menu.
```

The GRUB menu can be used to set kernel options at boot time.

There is also the human interface, system, processing, storage and networking subsystems.

While we are looking at the userspace interfaces, it is important to note that these layers should only change in very important circumstances. One of the golden rules of kernel development is that userspace (programs you interact with) should not be broken by changing kernel versions. While this could be limiting for the kernel developers, imagine what would happen if your favourite apps had to be patched, recompiled and redistributed every time your kernel was updated! The design of the ABIs (application binary interfaces) is incredibly important, so that functionality can be added over time, while retaining backwards compatibility.

Drivers and modules:

While Linux distros do need drivers to interact with the hardware, this is not handled in the same way as with Windows. The Linux kernel comes compiled with a huge number of devices. While the kernel gets regular releases, updates and support for new hardware, it does take time to get devices added. While this can be frustrating, it means that once the device has been added to the various source code trees, it is well tested before being made available. Where support for hardware devices is not built-in, it can be added using what is known as a kernel module. This is a piece of the kernel that has not been built as part of the standard release, but can be added on at runtime. Modules can be added at boot time or while your distro is running.

As is the case with setting kernel options, we are not giving specific examples here as the kernel is so

The main man himself – Linus Torvalds.

QUICK TIP

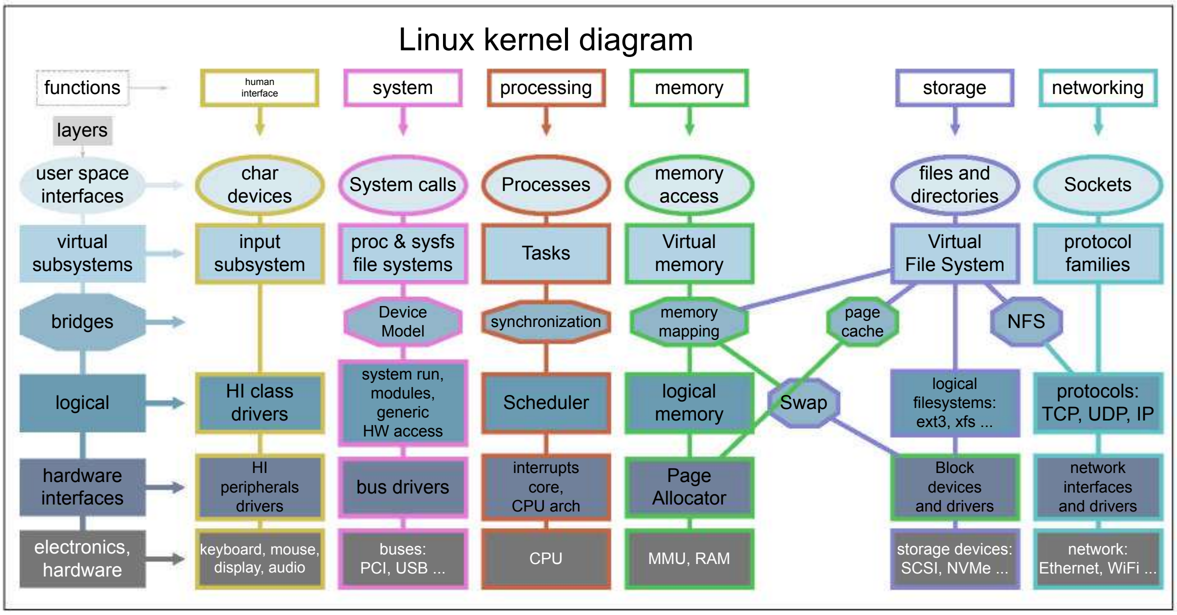
As it's all open source, the complete Kernel code can be downloaded from https://kernel.org.

SPEAKING THE SAME LANGUAGE

Given that the kernel is interfacing with hardware, it needs to be able to interact on a very low level. This involves being able to directly interact with memory, interrupts and various other concepts. When Linus Torvalds started writing the Linux kernel, he decided to use the venerable C language. Once compiled, C code runs very quickly and has full control of the memory with which it works. Where this can fall down, however, is that if the memory isn't handled correctly, lots of problems, such as memory leaks and security bugs, can ensue. Assembly is the other language currently used in the Linux kernel.

Starting in 2015, one of Mozilla's engineers started a personal project, a new programming language called Rust (see **LXF210-214**). Mozilla sponsored the project in 2019 and it is now managed by the Rust Foundation. Rust's popularity has exploded in recent years, partially due to its suitability for different programming tasks, focus on memory safety and high performance.

Rust has been adopted by the team behind the Linux kernel and work is ongoing to allow developers to write modules/drivers using it. Support is currently experimental, but if the language was completely unsuitable, this experiment would not have been started.



The Linux kernel split into its constituent parts. When visited in a web browser, the diagram is interactive and contains links to other sources.

CREDIT: https://github.com/makelinux/

QUICK TIP

An interactive diagram of the kernel can be viewed at https://makelinux.github.io/kernel/diagram/.

fundamental to the running of our operating system. These commands can be run from a terminal:

To load a single module:

```
$ sudo modprobe OPTION_NAME=VALUE MODULE_NAME
```

To load two modules:

```
$ sudo modprobe -a MODULE1 MODULE2
```

This unloads a single module:

```
$ sudo modprobe -r MODULE_NAME
```

These commands only add the module for the current boot. Adding the module name and options to a file in the **/etc/modules-load.d/moduleName.conf** file causes the kernel to load the specified module at each boot. Using a **blacklist-moduleName.conf** file allows a module to be blacklisted, which means it is not loaded, even if your distro thinks it needs to be. This may be required when modules cause each other problems if loaded together, or if multiple modules provide the same functionality and the automatically selected one isn't as good as another. The files in this path are used by *Systemd* to load modules at boot. An alternative directory at **/etc/modprobe.d/moduleName.conf** is used by *udev* to load modules at boot as well.

Dynamic Kernel Module Support or DKMS is a system used to build kernel modules dynamically from source code. This is useful in situations where the kernel module source code is stored outside the kernel source tree, and whenever the kernel is updated, the DKMS software is used to build the new version of the module and load it, once compiled.

Boot options

In the bad old days, it was often found that options needed to be tweaked when booting a Linux distro. Two different options that were often the culprit were those for ACPI, Advanced Configuration and Power Interface, and APIC, Advanced Programmable Interrupt Controller. Both of these configure how the kernel interfaces with some of the low-level hardware and often needed to be tweaked. To do so, when you see the GRUB bootloader, select the option you wish to change and select the `e` key. You may need to press the Tab key early in the boot process for the menu to show. You can scroll to the end of the line and add extra options or change existing ones. The magic incantation that was used a lot was `noapic acpi=off`.

This procedure sets the options for this boot only. To set it permanently, while booted into the Linux distro, edit the **/boot/efi/loader/<Distro Name>.conf** file and change the `options root=UUID=5b7be8ae-549d-4bb0-b85c-06d62b504709 ro quiet loglevel=0 systemd.show_status=false splash` line. Save the file and reboot. Ensure the line is edited correctly or you may find that your distro doesn't boot. These instructions are for Pop!_OS 22.04 and other distros may use a different bootloader, which would need different steps to follow.

Setting kernel options

It is also possible to change kernel options from a running Linux distro – they can be set temporarily or permanently. Examples are not shown here as options do not often need to be set and changing the incorrect one could cause issues. The following commands can be used to show all currently set options, set an option temporarily, and set the value so it's used each boot:

To view all the options:

```
$ sudo sysctl -a
```

Setting options temporarily:

```
$ sudo sysctl SYSTEM.AREA.OPTION=VALUE
```

To set options permanently, by writing the values to the necessary config file:

```
$ sudo sysctl -w SYSTEM.AREA.OPTION=VALUE >> /etc/sysctl.conf
```

BSD kernel

BSD is known for being incredible stable and probably behind in terms of the devices it supports. BSD is a great choice for appliances like routers and NASes. A fundamental difference between BSD and Linux systems is the development methodology. In BSD, a single source tree is used to develop the system, making it a lot easier to develop a cohesive system.

GNU Hurd

Hurd is a distribution of software tools with the Mach kernel and Hurd is the name of a set of daemons (services) and protocols designed to communicate with the Mach microkernel. Working together, these components can behave in the same way as the Linux kernel and can form part of a working OS.

A bit of fun

Let's have some fun. For all us ThinkPad-loving geeks, it is possible to switch on and off the LED on the lid. We could use Morse code to spell out a word or blink the LED when certain conditions are met. Sadly, the LED can't be blinked quickly enough to provide an idea of the beat of a song. To carry out the steps below, open your terminal and run the following commands:

```
$ sudo modprobe -r ec_sys
$ sudo modprobe ec_sys write_support=1
$ echo -n -e "\x0a" | sudo dd of="/sys/kernel/debug/
ec/ec0/io" bs=1 seek=12 count=1 conv=notrunc 2> /
dev/null
$ echo -n -e "\x8a" | sudo dd of="/sys/kernel/debug/
ec/ec0/io" bs=1 seek=12 count=1 conv=notrunc 2> /
dev/null
$ echo -n -e "\xca" | sudo dd of="/sys/kernel/debug/
ec/ec0/io" bs=1 seek=12 count=1 conv=notrunc 2> /
dev/null
```

In this code snippet, we are first of all removing the kernel module called ec_sys. We then load it again, but this time we set the **write_support** flag to the value of **1**, which allows us to write values to the devices the driver controls. The next three lines write to the same device, but the different values being written control what happens to the LED. The three lines switch off, switch on and blink the LED respectively. We will now step through the command to switch off the LED.

The value of **\x0a** is first of all written to the standard input of the *dd* command. This means that the data being fed in effectively replaces the **if** or input file argument of the *dd* command. The *dd* command is then used to write the value to the file, which is detailed by the **of** or output file argument. The other arguments of *dd* mean the following: the **bs=1** argument instructs *dd* to copy a single byte at a time; the **seek=12** argument writes to the file from the twelfth block in; and the **count=1** argument instructs *dd* to write a single byte. Finally, the **conv=notrunc** argument sets *dd* to not truncate the output file.

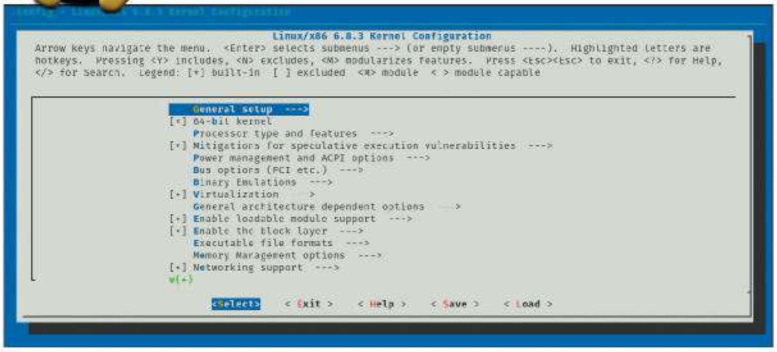
Compiling a kernel:

In the bad old days, if support was missing for a certain piece of hardware, it may have been necessary to compile your own kernel. This isn't as dreadful as it sounds and you can experiment by opening a terminal and using the instructions below:

```
$ sudo apt install build-essential libncurses-dev
bison flex libssl-dev libelf-dev fakeroot
```

Now navigate, in your browser of choice, to **http://kernel.org** and download the tarball of the version of the kernel that you would like to install. Once the file has downloaded, navigate to where it has been saved and decompress it. Once done, navigate back to your terminal and navigate to where the file is stored.

```
$ cd KERNEL_SOURCE_LOCATION
$ cp -v /boot/config-$(uname -r) .config
$ make localmodconfig
$ make menuconfig
$ fakeroot make -j8
```



```
.config - Linux/x86 6.8.3 Kernel Configuration
                         Linux/x86 6.8.3 Kernel Configuration
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are
  hotkeys.  Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help,
  </> for Search.  Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

                 General setup  --->
             [*] 64-bit kernel
                 Processor type and features  --->
             [*] Mitigations for speculative execution vulnerabilities  --->
                 Power management and ACPI options  --->
                 Bus options (PCI etc.)  --->
                 Binary Emulations  --->
             [*] Virtualization  --->
                 General architecture dependent options  --->
             [*] Enable loadable module support  --->
             [*] Enable the block layer  --->
                 Executable file formats  --->
                 Memory Management options  --->
             [*] Networking support  --->
             v(+)

                   <Select>    < Exit >    < Help >    < Save >    < Load >
```

The make menuconfig command is used to configure which parts of the kernel/modules to build.

When running these commands, we first of all copy a sample config file to give us something to work from. If we were to run the final command now, the building stage would take a long time. This is because a huge number of modules will also be compiled and included. By running the `make localmodconfig` command, the system is probed to determine which modules are currently being used. The `make menuconfig` loads an ncurses program, which can be used to further configure the kernel that you are going to build. Finally, `fakeroot make -j8` builds the kernel, using eight CPU cores. The number of cores can be adjusted.

Once the build has completed, `echo $?` can be used to determine if the build was successful or not. If it was, you can install the newly built kernel by running the following commands:

```
$ sudo make modules_install
$ sudo make install
```

If the build was unsuccessful, the following link contains items that can be changed and may allow the kernel to build: **https://davidaugustat.com/linux/how-to-compile-linux-kernel-on-ubuntu**.

The system can then be rebooted. Note: this is best experimented with by using a virtual machine, rather than on your daily driver as no one likes downtime.

We hope you have enjoyed this introduction to the Linux kernel and have a greater appreciation of the work the developers do, which enables us to use our computers in the way we do and allows Linux to scale from a phone to a router to a supercomputer. LXF

QUICK TIP

You can learn more about kernel architectures at www.techtarget.com/searchdatacenter/definition/kernel

» KEY PLAYERS

As mentioned, Linus Torvalds started the kernel in 1991, while he was at university. As time has gone by and the code base has mushroomed to support more and more hardware, each subsystem now has its own maintainers, who are responsible for this area of the code, before changes are considered to be merged with the kernel itself. Greg Kroah-Hartman is the maintainer of a number of subsystems as well as the stable version of the kernel. Greg is the de facto deputy to Linus Torvalds. Theodore T'so is also a prolific hacker and has been involved with kernel development for many years. Ted has worked in many areas, including the **/dev/random** device to generate random numbers, and filesystem utilities for the ext2, ext3 and ext4 filesystems, and is also a maintainer for the ext4 filesystem. Alan Cox is a programmer from the UK who was heavily involved in the kernel for a number of years and maintained the version 2.2 branch. A number of well-known open source companies employ developers to work on the kernel. These include Intel, Red Hat, Linaro, IBM, Samsung, SUSE, Google and AMD.

PERIODIC-TABLE-CLI

Credit: https://github.com/spirometaxas/periodic-table-cli

Learn the periodic table

An interactive periodic table accessible from the CLI might sound like sci-fi, but is pure joy for relapsed chemistry nerds like **Shashank Sharma**.

OUR EXPERT

Shashank Sharma is a trial lawyer in New Delhi and an avid Arch user. He's been writing about open source software for 20 years, and lawyering for 10.

It might sound we're joking, but if you think about it, the periodic table is one of the most easily recognised icons in the world. Certainly on a par with lab coats, stethoscopes, thermometers and Lego pieces. Anyone who looks at it knows exactly what it is. Some people are even required to memorise it as part of their studies, while others still quote it as a display of their nerd quotient. Although you could quickly do an online search to look up elements should the need ever arise, offline utilities have the advantage of offering pertinent information without overloading you with data.

Released under the MIT licence, *Periodic-table-cli* is an interactive utility that can help you master the 118 elements known to man. In addition to the Node.js offering discussed on the project's GitHub page, the developer also publishes a Python alternative.

Because of its esoteric nature, you understandably won't find *Periodic-table-cli* in the software repositories of desktop distributions. However, if your distro is already configured for Node.js or Python packages, you can use the `npm install –global periodic-table-cli` or `pip install periodic-table-cli` command to install it.

» CLI WORDLE

The terminal is a vastly robust resource that many use to accomplish a variety of tasks. In addition to listening to music, manipulating PDF files, reading emails, and editing image and video files, you can also learn how to type better, perform security audits, work with encrypted files, and much more, as discussed by this author over the past several years. But if you ever need a break from all that, you can also play games such as *Wordle* from the comfort of the CLI.

Written in C and released under the MIT licence, *Cwordle* is a nifty utility that's endless fun for anyone who's tired of hangman and is ready for an alternative dictionary game.

Clone the git repository with the `git clone https://github.com/velorek1/cwordle.git` command and navigate into the downloaded **cwordle/** directory. Next, run `cmake .` followed by the `sudo make install` command to install the word game.

You can now launch *Cwordle* with the `cwordle` command. When you type in your guess word, if a letter is in the correct position, it's highlighted in green. If the letter occurs in the word but is in the incorrect position, it's marked in orange. You have six attempts at guessing the correct word. If you fail, *Cwordle* displays the correct word and then quits.

Start learning

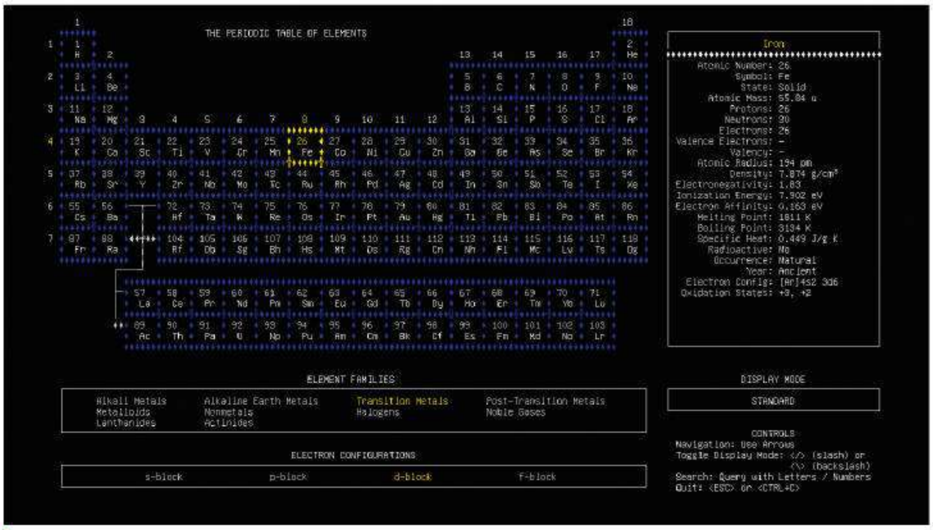
The periodic table is more than just a collection of elements. Each element has a distinct name and symbol. And each element has an atomic number that denotes the number of protons in its nucleus. The elements in the periodic table are listed in the order of increasing atomic number.

Depending on your terminal emulator utility and screen size, you might not be able to view all the information on display with *Periodic-table-cli*. This is because the tool requires a terminal screen capable of showing 156 columns and 46 rows. Your only recourse is to switch to full-screen mode, or scroll to access all the displayed information *(see Quick Tip)*.

When you run the tool with the `periodic-table-cli` command, all the elements in the periodic table are listed on the screen. The currently selected element is shown with a golden boundary. You can view the name of the selected element and read about it in the information box on the right. This column comprises myriad information such as Atomic Number, Symbol, State (solid, liquid or gas), Atomic Mass, Melting/Boiling Point, Year Of Discovery, and so on. In all, the utility displays a total of 22 data points in the information box for the selected element.

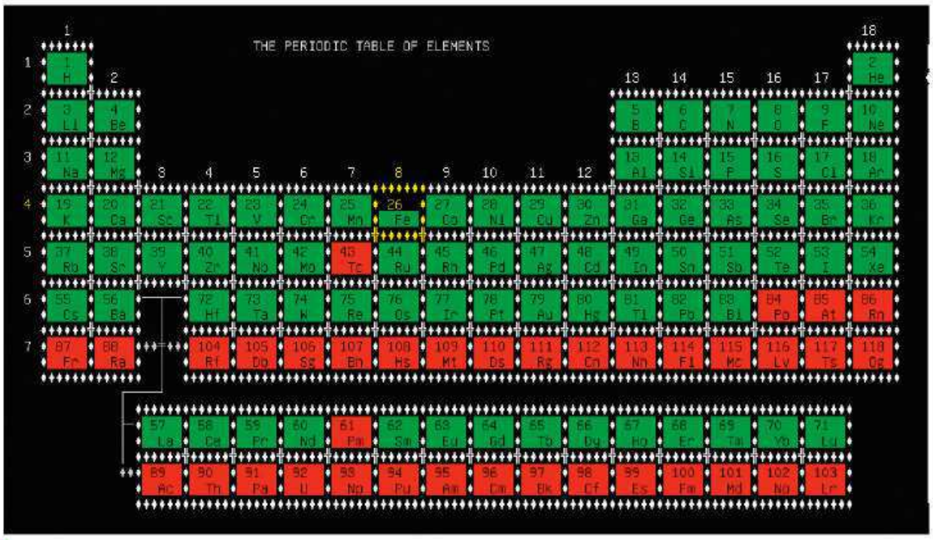
You can use the up/down arrow keys to scroll through the periodic table. At the bottom of the table, the utility provides two boxes. The first is a list of Element Families, such as Alkali Metals, Alkaline Earth Metals, Nonmetals, Noble Gases and so on. All the elements in the periodic table belong to one of these families. You can use the arrow keys to select a family, and then read about it in the information box.

When you select an item in the Element Families section, all elements in the periodic table belonging to the family are highlighted in the same colour. For instance, when you select Alkaline Earth Metals, Group 2 of the periodic table is highlighted. Similarly, all seven elements in Group 17 of the periodic table are highlighted when you select Halogens in the Element Family box. You'll find a list of Electron Configuration at the bottom of the interface. For each selected element,



```
THE PERIODIC TABLE OF ELEMENTS

                Iron
      Atomic Number: 26
             Symbol: Fe
              State: Solid
        Atomic Mass: 55.84 u
            Protons: 26
           Neutrons: 30
          Electrons: 26
  Valence Electrons: -
            Valency: -
      Atomic Radius: 194 pm
            Density: 7.874 g/cm³
  Electronegativity: 1.83
  Ionization Energy: 7.902 eV
  Electron Affinity: 0.163 eV
      Melting Point: 1811 K
      Boiling Point: 3134 K
      Specific Heat: 0.449 J/g K
        Radioactive: No
         Occurrence: Natural
               Year: Ancient
    Electron Config: [Ar]4s2 3d6
   Oxidation States: +3, +2

                        ELEMENT FAMILIES
  Alkali Metals    Alkaline Earth Metals   Transition Metals   Post-Transition Metals
  Metalloids       Nonmetals               Halogens            Noble Gases
  Lanthanides      Actinides

                     ELECTRON CONFIGURATIONS
      s-block         p-block         d-block         f-block

DISPLAY MODE
STANDARD

CONTROLS
Navigation: Use Arrows
Toggle Display Mode: </> (slash) or
                     <\> (backslash)
Search: Query with Letters / Numbers
Quit: <ESC> or <CTRL+C>
```

Students and enthusiasts will appreciate all the info on offer and the different display modes that make learning fun.



```
THE PERIODIC TABLE OF ELEMENTS
```

Use a terminal emulator that supports 256 colours to take full advantage of the Periodic-table-cli utility.

the 22 data points are displayed in the box on the right, and the element family and electron configuration are highlighted in the two boxes at the bottom. When you select an element, such as arsenic, you can immediately tell that it belongs to the Metalloids Element Family and has the p-bloc Electron Configuration.

While the utility doesn't feature a dedicated search box, you can still search for elements in the periodic table. Just start typing – either numbers or letters – and the utility matches what you type against element names, symbols and atomic numbers, as well as family names and electron configuration. All elements matching your typed keywords are displayed in the information box. Use the up and down arrow keys to highlight the element you're interested in, and hit Enter to select it. For instance, when you type `i`, the utility lists 23 matching elements, but the list drops to only three when you type `ir`.

Display modes

We've already discussed how the *Periodic-table-cli* utility displays element families and four electron configurations. In addition to these, you can also separate elements in the periodic table on the basis of their state (solid, liquid or gas).

The *Periodic-table-cli* utility features 20 display modes that use colours to sort the different elements. You can press `/` to switch between the different display modes. By default, the utility launches in the Standard Mode, as denoted in the Display Mode box under the information box on the right of the interface.

Press `/` to switch to the Element Families display mode. You'll now find the elements grouped into 10 different colour-coded element families. Press `/` again to switch to the Electron Configuration display mode, where four different colours are used to denote the different electron configurations. Press `/` once again to have the elements sorted into three colours for solid, liquid or gas.

You can similarly press `/` to cycle through all available display modes, such as the heat mass or atomic mass, melting point and boiling point of elements. The Radioactivity display mode uses the colour red to denote elements that are radioactive and green for elements that are stable. The Occurrence display mode similarly uses different colours to separate natural, rare and synthetic elements in the periodic table. The Year mode uses a heat map, increasing from blue to red, to display the year in which the elements were discovered. Ancient elements, such as copper, mercury and aluminium are all displayed in white.

Small screen boy

You can run the `periodic-table-cli --mode=chart` command to launch the utility in non-interactive mode. This still requires the terminal emulator width to be 113 characters. If you need a smaller version still, use the **-s** command option: `/periodic-table-cli --mode=chart -s`.

Instead of navigating through the periodic table looking for a particular element, you can also launch the periodic table with the specified element selected. Run `periodic-table-cli --mode=chart --name=<name>`. You can also use the **--symbol=<symbol>** or **--atomic-number=<atomic number>** options instead of the name to select the element. For instance, use `periodic-table-cli --mode=chart --name=mercury` to have mercury selected in the periodic table, and `periodic-table-cli --mode=chart --symbol=FE` to show iron.

But before anything else, it might make sense to set up an alias if you don't want to repeatedly type `periodic-table-cli` to launch the utility. LXF

QUICK TIP

Unlike some utilities that mould the display based on screen size, *Periodic-table-cli* instead cuts off information if the screen size is small. Your only recourse is to scroll to access this data. Press full stop (`.`) to scroll down, comma (`,`) to scroll up, and `<` and `>` to move left and right respectively.

» ENHANCE YOUR TERMINAL-FU Subscribe now at http://bit.ly/LinuxFormat

LINUX BASICS

Getting comfortable with terminal control

Once upon a time, nervous **Nick Peers** was afraid of command-line interfaces. Not any more!

OUR EXPERT

Nick Peers has been playing around with computers for over 30 years, and has been dabbling with Linux for the best part of a decade.

There's no escaping the Linux terminal. This is your gateway to accessing Linux's command-line interface (CLI) or shell. While all operating systems have an underlying CLI, none are more visible than the terminal. If you're new to Linux, it's tempting to studiously avoid it, but there are very good reasons for at least acclimatising yourself to the basics of Bash, the language used to power the Linux shell.

The terminal is the app used to access the shell. The good news is that it works in exactly the same way, so whatever you learn here will be helpful should you find yourself confronted by the command-line shell instead of your desktop due to some problem or other. What might be a daunting prospect is a little less frightening when you have some previous experience to draw on. But it's not simply about equipping yourself with some potentially vital Linux-saving skills, because another reason to love the terminal is that it's actually quicker to do certain things via a few quick-fire commands than wading through a series of point-and-click dialogs.

With this in mind, let's introduce you to how the terminal works, how to access it and how to get started with the fundamentals of using it.

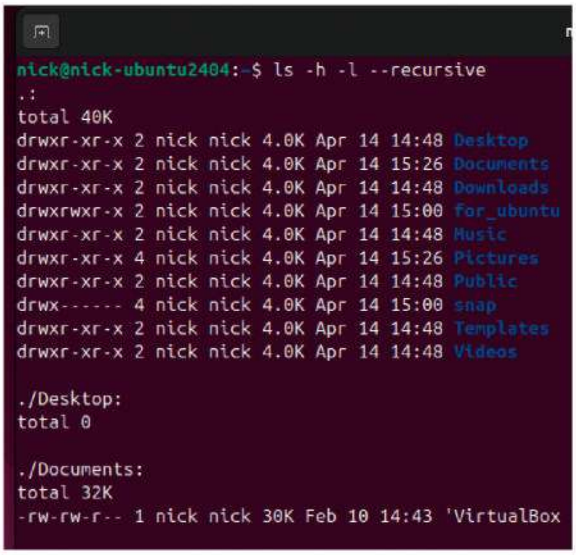
First steps

All Linux distros ship with a default terminal app – some even come with more than one. They all work in virtually the same way, so to keep things simple, we're using Ubuntu's terminal here. You can open it via the Launcher, or save time by pressing Ctrl+Alt+T. When opened, you should see something like the following:

```
user@pc-name:~$
```

The line is colour-coded – the first section marked in green (`user@pc-name`) confirms you're logged in to the terminal on your own PC under your own user account (such as **nick@ubuntu2404**). Next to this is a blue tilde (~) symbol, which reveals the terminal's current working directory. Because the terminal always opens to your personal **Home** folder, it displays the ~ shortcut. Finally, the white **$** symbol is where your typed commands will appear. We use the **$** to indicate terminal commands in *Linux Format* – type everything after the **$** and press Enter to execute the command.

When you log into the terminal as your own user, you don't have administrator – or root – access to your system for security purposes. This means you only have limited access to certain commands. If you've come from Windows or Mac OS, you'll know that when you attempt certain functions, such as installing new software, you're prompted to enter your user password or click a dialog to continue. This basically grants your user temporary administrator access to allow it to perform system-related tasks.



```
nick@nick-ubuntu2404:~$ ls -h -l --recursive
.:
total 40K
drwxr-xr-x 2 nick nick 4.0K Apr 14 14:48 Desktop
drwxr-xr-x 2 nick nick 4.0K Apr 14 15:26 Documents
drwxr-xr-x 2 nick nick 4.0K Apr 14 14:48 Downloads
drwxrwxr-x 2 nick nick 4.0K Apr 14 15:00 for_ubuntu
drwxr-xr-x 2 nick nick 4.0K Apr 14 14:48 Music
drwxr-xr-x 4 nick nick 4.0K Apr 14 15:26 Pictures
drwxr-xr-x 2 nick nick 4.0K Apr 14 14:48 Public
drwx------ 4 nick nick 4.0K Apr 14 15:00 snap
drwxr-xr-x 2 nick nick 4.0K Apr 14 14:48 Templates
drwxr-xr-x 2 nick nick 4.0K Apr 14 14:48 Videos

./Desktop:
total 0

./Documents:
total 32K
-rw-rw-r-- 1 nick nick 30K Feb 10 14:43 'VirtualBox
```

Get a detailed breakdown of your folders (and their contents) using the ls tool in combination with the right flags.

QUICK TIP

Use the *Whatis* tool to quickly get a summary of any tool you enter after it – for example, `whatis apt` . For a more detailed explanation of a tool, including all its supported commands and flags, use *Man* instead: `man apt` .

The same is true with parts of the Linux desktop, and it applies to the terminal, too. The question is, how do you grant yourself that temporary access using Bash? The simple answer is to prefix commands using `sudo` – but for a more detailed explanation, check out the How To Gain Root Access box *(opposite).*

Terminal commands

We've jumped ahead of ourselves slightly. Before going further, let's explain how Bash works. It's both a shell and a command language, and its syntax is very simple:

```
$ utility <command> -options
```

The **utility** section of the command is the tool you wish to run, such as *ls* for listing the contents of a directory, *cd* for changing to a different directory, or *apt* to open the package management tool. Next to this is **command** – which specifies what you want the tool to do – and **-options**, which enables you to set one or more options using flags. Each flag can be preceded by one (-) or two (--) dashes, and the simplest one to remember is **--help**, which provides a brief description of the utility, plus lists all commands and options the tool can use:

```
$ apt --help
```

Some utilities can be run without any commands or options – `ls` on its own provides a basic list of all folders and files in a directory, for example. You can also run utilities with a combination of commands and/or options – for example:

```
$ ls ~/Documents -l
```

This instructs *ls* to list the contents of your personal **Documents** folder. Adding the **-l** flag instructs *ls* to provide more detailed information about its contents.

Files and folders

When you enter the terminal, you're placed in what's known as your working directory. While many tools can be run regardless of which folder you're in, it can help to sometimes be in a specific folder, particularly when working with files or folders using utilities like *mkdir* (create folder) or *rm* (delete files and/or directories).

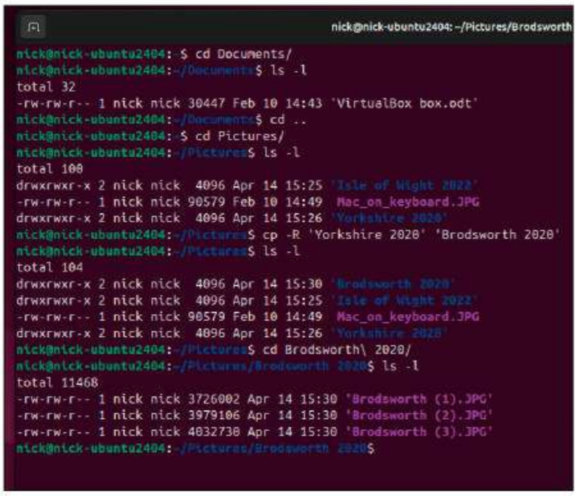
As we said above, every time you open a terminal window, you're placed in your **Home** directory by default, which is the **/home/username** folder on your hard drive, but represented as ~ in the terminal as a kind of shorthand.

To confirm this, type `ls` and hit Enter. You'll see a list of all folders (in blue) and files (in white) in your **Home** folder. These should include all your user folders like **Documents**, **Desktop** and **Pictures**, confirming you're in your **home** folder. Now type `ls -l` and hit Enter, and you'll also see the date and time the folder or file was last modified, its size in bytes, the user and group who own that folder or file, and user permissions displayed as a series of text like `drwxrw-r--`. For more on user permissions, check out the box *(over the page)*.

Other flags worth considering include **-a**, which displays all files, including those that begin with a period mark (.) that are normally hidden from view. There's also **--recursive**, which lists files recursively. This basically means that the contents of any subfolders such as **Documents** are also displayed.

You can also combine flags – for example, *ls's* **-l** flag lists file sizes in bytes, which isn't the most practical option, so you can also add the **-h** flag, like so:

```
$ ls -h -l
```



```
nick@nick-ubuntu2404: ~/Pictures/Brodsworth
nick@nick-ubuntu2404:~$ cd Documents/
nick@nick-ubuntu2404:~/Documents$ ls -l
total 32
-rw-rw-r-- 1 nick nick 30447 Feb 10 14:43 'VirtualBox box.odt'
nick@nick-ubuntu2404:~/Documents$ cd ..
nick@nick-ubuntu2404:~$ cd Pictures/
nick@nick-ubuntu2404:~/Pictures$ ls -l
total 100
drwxrwxr-x 2 nick nick  4096 Apr 14 15:25 'Isle of Wight 2022'
-rw-rw-r-- 1 nick nick 90579 Feb 10 14:49  Mac_on_keyboard.JPG
drwxrwxr-x 2 nick nick  4096 Apr 14 15:26 'Yorkshire 2020'
nick@nick-ubuntu2404:~/Pictures$ cp -R 'Yorkshire 2020' 'Brodsworth 2020'
nick@nick-ubuntu2404:~/Pictures$ ls -l
total 104
drwxrwxr-x 2 nick nick  4096 Apr 14 15:30 'Brodsworth 2020'
drwxrwxr-x 2 nick nick  4096 Apr 14 15:25 'Isle of Wight 2022'
-rw-rw-r-- 1 nick nick 90579 Feb 10 14:49  Mac_on_keyboard.JPG
drwxrwxr-x 2 nick nick  4096 Apr 14 15:26 'Yorkshire 2020'
nick@nick-ubuntu2404:~/Pictures$ cd Brodsworth\ 2020/
nick@nick-ubuntu2404:~/Pictures/Brodsworth 2020$ ls -l
total 11468
-rw-rw-r-- 1 nick nick 3726002 Apr 14 15:30 'Brodsworth (1).JPG'
-rw-rw-r-- 1 nick nick 3979106 Apr 14 15:30 'Brodsworth (2).JPG'
-rw-rw-r-- 1 nick nick 4032730 Apr 14 15:30 'Brodsworth (3).JPG'
nick@nick-ubuntu2404:~/Pictures/Brodsworth 2020$
```

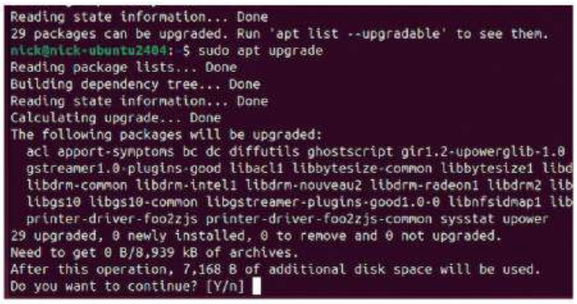
The recursive (-R) flag instructs Linux to copy not just the named folder, but all its contents (files and subfolders), too.

This lists file sizes in kilobytes, megabytes or even gigabytes, depending on their size. Don't forget to use **--help** to find out how else you can configure *ls*.

Navigate your files

We use *cd* to move between folders. To move into a folder inside the current working directory, type `cd` then the name of the subfolder, such as `cd Downloads`. Folder and filenames are case-sensitive, so if the folder begins with a capital letter, you're told `No such file or directory` if you type `cd downloads` by mistake.

You can also move down several levels at once using the following syntax: `cd subfolder/subfolder2`. To move back up to the previous level, use `cd ..` – you can also use the `/` character to move up multiple levels at once – for example, `cd ../..` moves up two levels.



```
Reading state information... Done
29 packages can be upgraded. Run 'apt list --upgradable' to see them.
nick@nick-ubuntu2404:~$ sudo apt upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages will be upgraded:
  acl apport-symptoms bc dc diffutils ghostscript gir1.2-upowerglib-1.0
  gstreamer1.0-plugins-good libacl1 libbytesize-common libbytesize1 libd
  libdrm-common libdrm-intel1 libdrm-nouveau2 libdrm-radeon1 libdrm2 lib
  libgs10 libgs10-common libgstreamer-plugins-good1.0-0 libnfsidmap1 lib
  printer-driver-foo2zjs printer-driver-foo2zjs-common sysstat upower
29 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 0 B/8,939 kB of archives.
After this operation, 7,168 B of additional disk space will be used.
Do you want to continue? [Y/n] █
```

The apt package management tool can quickly update your system as well as programs installed through Ubuntu's repositories.

What if you want to go somewhere completely different? Use `cd /` to place yourself in the root

QUICK TIP

Speed things up by including multiple commands on the same line. Use the && separator like an AND command – the second command runs only if the first is successful like so: `sudo apt update && sudo apt upgrade`.

» HOW TO GAIN ROOT ACCESS

There are two ways to gain root access to your system. Most Linux systems, including Ubuntu and Mint, preinstall the **sudo** package, which allows you to run administrator-level tasks simply by prefixing the command with `sudo` – for example, `sudo apt update`. You are prompted for your user password the first time you use **sudo** in each session, and occasionally thereafter if you spend a long time in the terminal before using **sudo** again.

Not all distros ship with **sudo** – Debian is a case in point. Instead, Debian expects you to switch to the root user by typing the following and pressing Enter:

```
$ su
```

When prompted for your password, you need to enter the separate password you set up for the root user during setup before pressing Enter. The prompt then changes to the following:

```
root@pcname:/home/
username#
```

When you're done working as the root user, type `exit` and hit Enter. You can't use **su** in Ubuntu; instead, type `sudo -i` and hit Enter, entering your user password if prompted.

Once logged in as root under **su**, Debian users can then install **sudo** with the following command:

```
$ apt install sudo -y
```

»

QUICK TIP

If you plan to navigate to a folder directly after opening the terminal, save time by navigating to that folder in *Files*. Now right-click anywhere inside the folder window and choose Open In Terminal to open a new terminal window pointing directly to that location.

directory or jump to anywhere on your system by entering the exact path, including that preceding `/` character to indicate you're navigating from the top level – for example, `cd /media/username`. The `~` character works for your **home** directory in a similar way, so typing `cd ~/Documents` is the same as typing `cd /home/username/Documents`.

One final time-saving tip: if you've jumped to a completely different directory, but want to go back to the previous folder, type `cd -` and hit Enter.

Copy and move files

Two tools handle file management: *cp* for copying, and *mv* for moving. Both handle paths in the same way as *cd,* making it easy to copy or move files between two directories – you don't even need to be in the source or destination directory for them to work.

Both commands work in the same way: enter the tool followed by the source (the file you wish to copy or move) and the destination (the directory you want it to go). So, to copy a file from your **Downloads** folder to your **Documents** folder, you would type:

```
$ cp ~/Downloads/document.odt ~/Documents/
```

This creates a copy with the same filename as the original. If you want to rename the copied file:

```
$ cp ~/Downloads/document.odt ~/Documents/
newname.odt
```

If you use *mv* in place of *cp,* the file would be moved and then renamed. You can also use *cp* to create a backup of a file in the same folder:

```
$ cp document.odt doc-backup.odt
```

This syntax is also used with *mv* to rename a file:

```
$ mv document.odt newname.odt
```

One potential danger of this approach is that if there's another file with the same name as the one you're copying, moving, backing up or renaming, it's overwritten. To avoid potential data loss, get in the habit of including the **-i** flag. The i stands for interactive, so if another file exists with the same name, you get a warning, enabling you to abort the process:

```
$ mv -i document.odt newname.odt
```

More commands

Both *cp* and *mv* are used to copy and move folders, too. For this to work, you need to make use of another flag: **-r** for recursive:

```
$ cp -r ~/Documents /mnt/sdb1/Backup/
```

If the **Backup** folder exists, a copy of the **Documents** folder is created inside it, complete with all its files and subfolders; if not, then the **Backup** folder is created, and the contents of the **Documents** folder are copied into that instead.

Use the *rm* command to delete a single file – for example, `rm invoice.odt`. The *rmdir* command deletes folders, but only empty ones. So, if you want to delete a folder and all its contents, use the following command: `rm -r foldername`.

You can also create new folders with *mkdir* – simply type `mkdir folder`, replacing `folder` with your chosen folder name to create it in the current directory, or provide a full path to where you want it to go: `mkdir ~/Documents/Work`. Use the *touch* utility to create an empty file – such as `touch config.sys`.

Things get complicated with files and folders that contain spaces in their filenames, such as **Star Trek**. When navigating or creating these folders or files, you need one of two special characters: the single quotation mark (`'`) or the backslash (`\`).

Use single quotation marks to enclose files or file paths that contain spaces, such as:

```
cd '~\Documents\Star Trek'
```

You should also use quotation marks when creating folders in this way. For example, simply typing `mkdir Battlestar Galactica` creates two separate folders called **Battlestar** and **Galactica**; type `mkdir 'Battlestar Galactica'` to get the folder you actually want.

You can also use `\` to get around this – `mkdir Battlestar\ Galactica` works in the same way, because the `\` character instructs *mkdir* to treat the following character (in this instance the space) as special.

Managing packages

In next issue's Basics tutorial, we'll reveal the various ways of getting software packages installed on your system, from self-contained Snaps and portable AppImages to programs housed in traditional

» CHANGING FILE PERMISSIONS

File and folder permissions are a key part of Linux – go to **https://help.ubuntu.com/community/FilePermissions** for a primer. Use `ls -l` to reveal existing folder and file permissions and how they apply to three levels: the item's owner, a specific user group and then everyone else. Permissions are **r** (read access), **w** (write access) and **x** (execute access for programs).

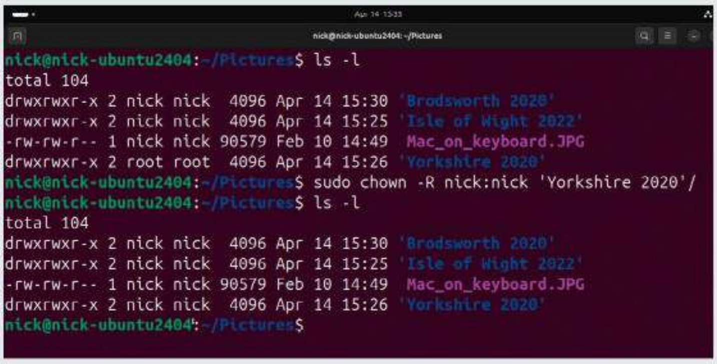
A file or folder's permission can be changed using the *chmod* tool, which assigns numbers rather than letters. Add `sudo` if you're not currently the file or folder's owner:

```
$ chmod 777 filename
```

When it comes to setting folder permissions, use the **-R** (recursive) option to apply the same permissions to all its contents (subfolders and files), so `chmod -R 777 folder`.

The key element here is the number, which refers to the permissions you're setting, with the first digit for the file owner, the next the file's group and final one for everyone else. These are typically one of: **4** (**r--**, read only), **5** (**r-x**, read and execute), **6** (**rw-**, read/write) and **7** (**rwx**, full access). So, **644** gives the owner read and write access, but restricts everyone else to read-only access. Set **775** and the owner and named group have full access, while everyone else can read and execute the file only.

A related tool to *chmod* is *chown,* which is used to change the owner or group of a file or folder. Use `sudo chown user:group file` to change the user and group that owns it. Again, the **-R** flag can be used with folders.



```
nick@nick-ubuntu2404:~/Pictures$ ls -l
total 104
drwxrwxr-x 2 nick nick  4096 Apr 14 15:30 'Brodsworth 2020'
drwxrwxr-x 2 nick nick  4096 Apr 14 15:25 'Isle of Wight 2022'
-rw-rw-r-- 1 nick nick 90579 Feb 10 14:49  Mac_on_keyboard.JPG
drwxrwxr-x 2 root root  4096 Apr 14 15:26 'Yorkshire 2020'
nick@nick-ubuntu2404:~/Pictures$ sudo chown -R nick:nick 'Yorkshire 2020'/
nick@nick-ubuntu2404:~/Pictures$ ls -l
total 104
drwxrwxr-x 2 nick nick  4096 Apr 14 15:30 'Brodsworth 2020'
drwxrwxr-x 2 nick nick  4096 Apr 14 15:25 'Isle of Wight 2022'
-rw-rw-r-- 1 nick nick 90579 Feb 10 14:49  Mac_on_keyboard.JPG
drwxrwxr-x 2 nick nick  4096 Apr 14 15:26 'Yorkshire 2020'
nick@nick-ubuntu2404:~/Pictures$
```

Use chown with care – if you set the wrong path, you could screw up your filesystem.

repositories (known as repos). Ordinarily, you'd use Ubuntu's App Center to manage software, but this increasingly favours Snap packages over those installed through Ubuntu's own repos.

Thankfully, the terminal offers *apt*, a package management tool focused exclusively on repo-housed packages. Let's start with the basics:

```
$ sudo apt update
```

This downloads the latest package lists from all listed repositories. You're notified if updates are available, and these can be installed in one go with:

```
$ sudo apt upgrade
```

You can also use *apt* to install new packages:

```
$ sudo apt install packagename
```

This installs the named package. If it requires other packages (known as dependencies) to run that aren't installed, *apt* downloads and installs these, too.

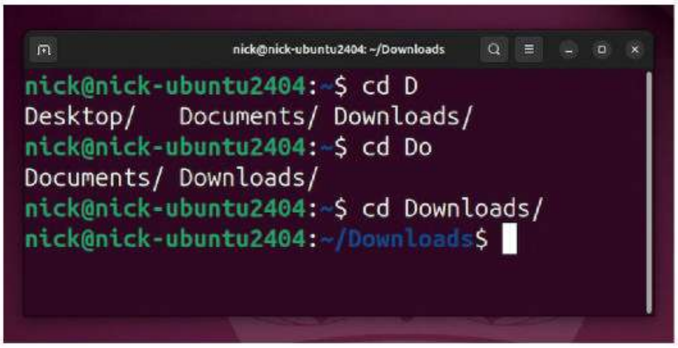
Naturally, *apt* can uninstall unwanted packages, too:

```
$ sudo apt remove packagename
```

Use `sudo apt purge packagename` to remove any configuration files along with the package, and `sudo apt autoremove` to free up disk space by removing packages installed by other packages that are no longer being used. Other useful *apt* commands include `apt check`, a diagnostic tool that checks for broken dependencies, and `apt autoclean`, which removes DEB files from removed packages.

We've scratched the surface of what you can do with the terminal, so visit **www.linuxformat.com/archives** to see what other tutorials and features we offer – **LXF222**'s Jedi Master The Terminal (**https://bit.ly/lxf222_feat**) feature is available to all users, while subscribers can access a whole host of tutorials by simply searching the full archive for 'terminal'.

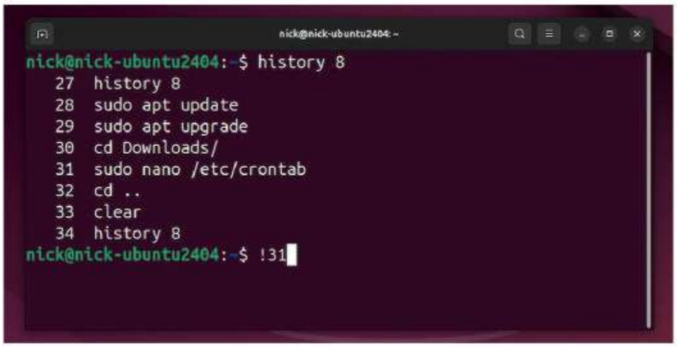
TIME-SAVING TIPS AND TRICKS



```
nick@nick-ubuntu2404:~$ cd D
Desktop/    Documents/ Downloads/
nick@nick-ubuntu2404:~$ cd Do
Documents/ Downloads/
nick@nick-ubuntu2404:~$ cd Downloads/
nick@nick-ubuntu2404:~/Downloads$ 
```

1 Tab autocomplete

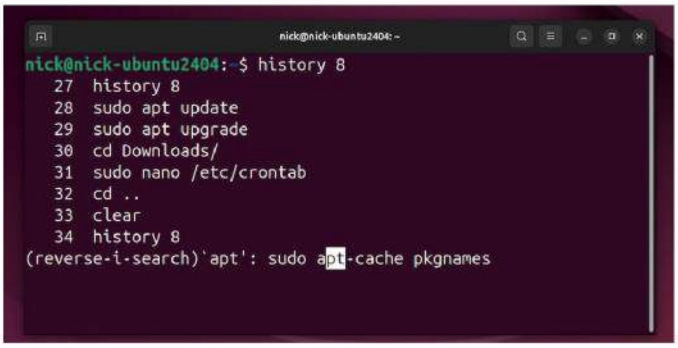
When typing in long path names or commands, try pressing Tab after typing a few letters – if only one suggestion exists, the terminal autocompletes the word or path or for you. However, if more than one suggestion exists, you need to press Tab twice to see the potential matches, and then add more characters to make your target the only possible choice before hitting Tab again.



```
nick@nick-ubuntu2404:~$ history 8
   27  history 8
   28  sudo apt update
   29  sudo apt upgrade
   30  cd Downloads/
   31  sudo nano /etc/crontab
   32  cd ..
   33  clear
   34  history 8
nick@nick-ubuntu2404:~$ !31
```

2 Repeat previous commands

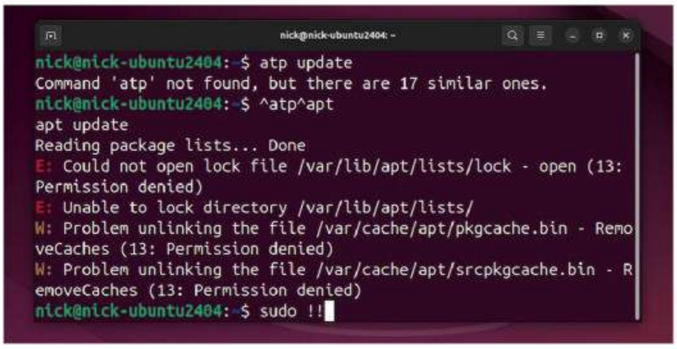
If you frequently repeat commands, you can save time with the help of the up arrow key. Press this to cycle back through your command history, and then press Enter to apply your chosen command. Alternatively, you can type `history` and press Enter to view your command history or `history 5` to view the last five commands. To repeat a command, type `!` followed by the number (for example, `!2`) and press Enter.



```
nick@nick-ubuntu2404:~$ history 8
   27  history 8
   28  sudo apt update
   29  sudo apt upgrade
   30  cd Downloads/
   31  sudo nano /etc/crontab
   32  cd ..
   33  clear
   34  history 8
(reverse-i-search)`apt': sudo apt-cache pkgnames
```

3 Search history

If you can't easily find a previous command, you can try a search for it: press Ctrl+R to search your history by keyword. As you type, the most recent command containing your search terms is displayed. Press Ctrl+R to keep cycling back to earlier matches. When you've found the command you want to repeat, just press Enter.



```
nick@nick-ubuntu2404:~$ atp update
Command 'atp' not found, but there are 17 similar ones.
nick@nick-ubuntu2404:~$ ^atp^apt
apt update
Reading package lists... Done
E: Could not open lock file /var/lib/apt/lists/lock - open (13:
Permission denied)
E: Unable to lock directory /var/lib/apt/lists/
W: Problem unlinking the file /var/cache/apt/pkgcache.bin - Remo
veCaches (13: Permission denied)
W: Problem unlinking the file /var/cache/apt/srcpkgcache.bin - R
emoveCaches (13: Permission denied)
nick@nick-ubuntu2404:~$ sudo !!
```

4 Fix mistakes quickly

Accidentally forgot to include `sudo`? Instead of laboriously typing out the entire command again prefixed by `sudo`, just type `sudo !!` and hit Enter to repeat the last command with `sudo` applied to it. And if you accidentally introduce a typo into a command, just use the following syntax to correct the mistyped word (in this case, correcting `atp` with `apt`): `^atp^apt`.

» WE PROMISE NO MORE TERMINAL! Subscribe now at http://bit.ly/LinuxFormat

BATOCERA

Credit: https://batocera.org/download

Playing a full (Steam) Deck of the classics

Christian Cawley explores retro gaming OS Batocera's Steam Deck build, and learns how to bring the classics to life (again).

OUR EXPERT

Christian Cawley is so obsessed with retro gear (particularly the 16-bit era) that the first game he installed on his Steam Deck was *Zool*, a 1992 Amiga platformer.

If you own a Steam Deck, you probably know that it's a remarkable handheld gaming PC that runs on a custom version of Arch Linux. With access (via an integrated Steam client) to a vast number of games released primarily on Windows, the Steam Deck can run just about any current game. This is thanks to a combination of factors, not least the Steam Proton compatibility layer, which (in general terms) uses *Wine* and DirectX over Vulkan (DXVK). The result is games that play great on the Steam Deck, giving you the option of taking those titles anywhere.

But what if you wanted to play some slightly older games?

Beyond the titles in the Steam library from the early 2000s, or installing an interface to access your GOG account, your best option is a retro-gaming suite with a build specifically for the Steam Deck, such as Batocera.

What is Batocera?

An open source and free-to-use retro-gaming distribution, Batocera can be copied to a USB stick or microSD card to turn a PC, SBC or game console into a retro-gaming system. The change can be temporary or permanent, and doesn't require any modification.

Linux is blessed with a panoply of retro-gaming options, from distros such as Lakka to Raspberry Pi-focused options such as RetroPie. Batocera is at least as good as these platforms, but is unique in offering a dedicated Steam Deck version.

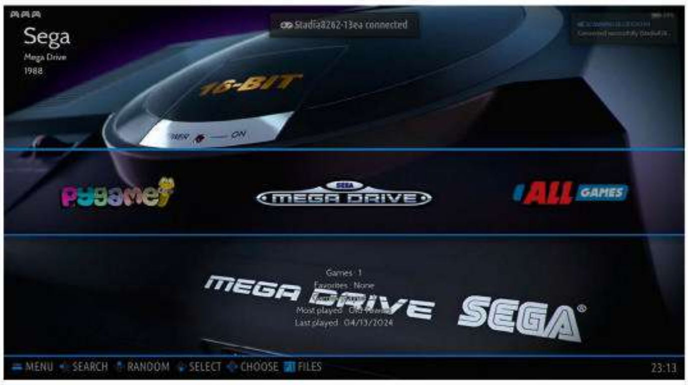
Built around a philosophy of 'download, flash, connect and play', Batocera has a massive selection of features. Over 190 systems can be emulated, features such as rewinding, display bezels, themes and shaders, and even achievements can be managed, and the software is completely free and open source.

Naturally, you need to own any games you play on Batocera – see our boxout *(page 59)* for a deeper explanation of this.

Just in case you don't have any retro games handy to play on Batocera, it comes with a nice selection of open source titles. You'll know some of them, as they are based on source code that was released to the public – titles such as *Doom*, and clones of games such as *Donkey Kong* and *Prince of Persia*.

QUICK TIP

You don't need a Steam Deck to enjoy Batocera emulation fun, there's a standard x86 PC build ideal for any spare laptop or PC you have.



Need to connect a Bluetooth controller? When a device is paired, a pop-up notification confirms the status.

As with other versions of Batocera, you don't need to install it on your Steam Deck. The software is bootable and can run from a microSD card. If you already use a microSD card for additional storage on your Steam Deck, you can simply swap them over.

Steam Deck time

Before you get started, ensure you have a good-quality microSD card. Specifically, look for a microSDXC UHS-I card and above. SanDisk media intended for the Nintendo Switch (U3 rated, and easy to spot thanks to a Nintendo mushroom) should be the minimum. SanDisk's Ultra and Extreme cards have the best reputation, and are available in 1TB models if needed.

To install Batocera, start by downloading the Steam Deck image from **https://batocera.org/download**. Then grab a copy of *Etcher* (**https://etcher.balena.io/#download-etcher**) and install this on your PC. It is available as an AppImage for 32-bit and 64-bit Linux. After downloading, set the file as executable: right-click the **balenaEtcher-xxx-x64.AppImage** file, then go to Permissions. Ensure all are set to Read and Write. Click Back, then enable Executable As Program. Close the window, then double-click to launch the AppImage.

If you prefer, you can download the DEB or RPM (**https://github.com/balena-io/etcher/releases/**) and install it from the terminal with *apt* or *yum:*

For Debian/Ubuntu PCs, use:

```
$ sudo apt install ./balena-etcher_[VERSION-NUMBER]_amd64.deb
```

For the Fedora family, install the file with:

```
$ sudo yum localinstall balena-etcher-[VERSION-NUMBER].x86_64.rpm
```

At this stage, insert the microSD card (with adaptor if required) into your Linux PC. With *Etcher* running, under Select Image click Flash From File to select the Batocera download. Don't worry about it being in the GZ format, as *Etcher* can unzip the disk IMG file within. Use the Select Drive option to choose your microSD card (internal drives should be hidden from this screen).

When ready, click Flash to write the Batocera disk image to your microSD card. After the process completes, close *Etcher*. The SD card may remain mounted, and you'll notice from your file manager that it has two partitions: **Batocera** and **Share**.

The next step is to eject the card in Linux, remove the physical media, and insert it into the Steam Deck.

Tweaking Batocera

To use Batocera, shut down your Steam Deck (press Steam, go to Power, then Shut Down). Once the screen goes off, hold the volume down button and press the power button until you hear the chime. Release both buttons and wait for the boot select screen to appear. The microSD card should be the second option in the list, labelled something like EFI SD/MMC Card

You can select this with the Steam Deck's usual controls, either the D-pad or the right touchpad, then A to confirm. (With subsequent boots, Batocera gets its own entry in the list. Should you wish to boot into the Steam Deck's usual environment, use EFI Hard Drive.)

Be sure to boot from a powered-off Steam Deck, as the process doesn't work if using the restart option. (Subsequent boots go directly into Batocera unless you repeat the above process to select the internal SSD.)

Once in the Batocera environment, pre-installed systems are accessible via a carousel menu. On the build downloaded for this guide, games for Sega Mega Drive/Genesis, NeoGeo, Commodore 64, Game Boy Advance and some open source clones were included.

Batocera is navigated using the Steam Deck's usual controls. The Select button opens the Quick Access menu for things such as shutdown, plane mode and the Batocera user manual. The Start button opens the Main menu, which gives access to everything from network and sound to controller and Bluetooth settings. You can even open the *Kodi* media centre.

Once a game is running, the controls change slightly. Here, the Select button is widely used. For example, a RetroArch menu can be opened using Select+A. Here, specific options for the emulator concerned can be found, controls/input can be tweaked, and screenshots, videos and streaming configured. While general shaders can be cycled through with Select+R2/L2, they can be enabled (and added) via the RetroArch menu. Other options include latency and rewind settings.

Meanwhile, you can use the Select+D-pad up and down buttons to select a save state slot, Select+X to save to the slot, and Select+Y to load from the slot.

On *MAME* and other emulated arcade game titles, the Select button inserts/adds credits, while the Start button starts the game.

» OTHER VERSIONS OF BATOCERA

Batocera's Steam Deck version has certainly raised awareness of the project, but it's been going for a few years. First released in 2016, Batocera has builds for a variety of different systems.

First, you've got its most popular versions, for desktop PCs, laptops, NUCs and Intel-based Apple systems. The Steam Deck version and Raspberry Pi 4 and 5 builds also fall into this ranking.

Sticking with the Raspberry Pi, Batocera is available for all consumer models, from the very first Pi onwards. This list includes Zero models (including the Retroflag GPi Case) and the CM3. Other boards such as the Libre Computer H5 and Le Potato, Odroids, Asus Tinkerbaord, Orange Pi 5 and ROCKPro64 can also run Batocera.

In addition, a bunch of handhelds besides the Steam Deck can run Batocera. These include the Odroid Go, Powkiddy and Anbernic devices. In addition, the Capcom Home Arcade two-player system also supports this retro-gaming platform.

However, note that any retro-gaming hardware that comes pre-installed with Android – such as the Retroid Pocket and related systems – cannot run Batocera.

Want to turn your older PC into a Linux-powered Batocera retro system? 32-bit CPUs and Intel Atom systems are also supported.

If you don't own a Steam Deck, you can check if your device is suitable for Batocera using the filter tool on the download page: **https://batocera.org/download**

User interface tweaks can be made, too, via User Interface Settings and Game Collection Settings.

Some of the controls on Batocera need some alteration for use on the Steam Deck. For example, exiting emulators or saving states is a bit hit and miss. Getting out of emulators can be a combination of different buttons. It should be Select+Start but you may find this combination needs to be pressed twice.

In case you accidentally end up in the *Dolphin* file manager from pressing the … button on your Steam Deck, use the left-hand touchpad to open the File menu and R5 as the left mouse button to exit.

Add a controller

If you're using the Steam Deck while docked, it's a good idea to use an external controller. USB devices are supported via Steam Deck docks (or USB hubs) but so, too, are Bluetooth controllers.

To connect a Bluetooth controller, hit the designated Start button, then Controller & Bluetooth Settings. In the next screen, choose Pair A Bluetooth Device Manually, and put your Bluetooth controller in

QUICK TIP

Copy your retro gaming ROMs to your the Share partition on the Batocera microSD card. While some ROMs just work, system BIOS files may be required to unlock some of the 150-plus emulators.

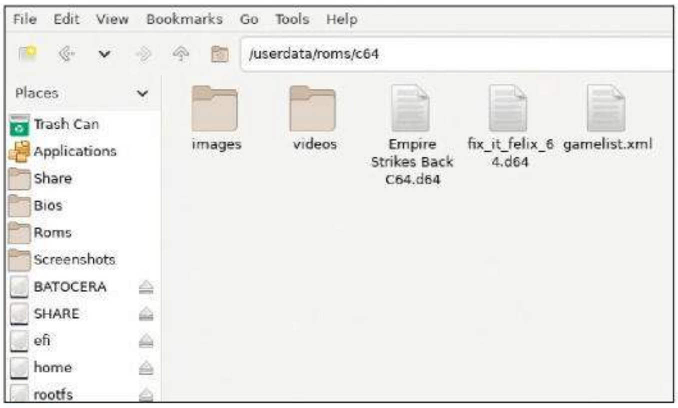
You can emulate games from any era or any platform on your Steam Deck, thanks to the emulators in Batocera.

»

ADD GAME ROMS TO BATOCERA

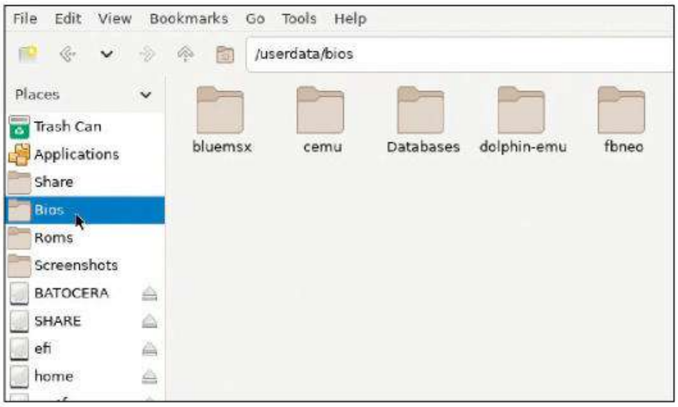
1 Copy ROMs to USB

To get your collection of retro-gaming ROMs on to your Steam Deck and playable through Batocera, the easiest option is to use a USB stick. If you don't have a dock or USB hub, you need to use a USB-C to USB-A adaptor to connect the USB device to your Steam Deck.



2 Use the corresponding folders

With a mouse connected to your Steam Deck (USB or Bluetooth), press the ellipsis button (below the right touchpad) to open the *Dolphin* file browser. Here, you should see the USB device attached. Open this to copy the contents into the corresponding **roms/** subdirectories in the **Shared** partition.

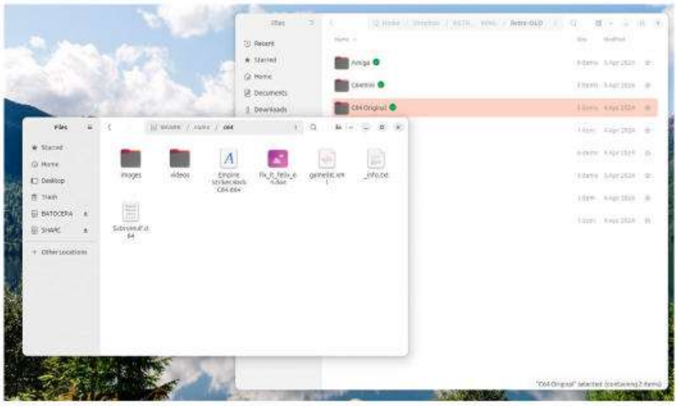


3 BIOS file locations

If you need to add BIOS files for your favourite emulators, instead of copying to the corresponding **roms/** folder, copy to **bios/** instead. Note: not all platforms need BIOS files. Problems can be detected by switching back into the EmulationStation interface (go to File > Close Window) and testing.

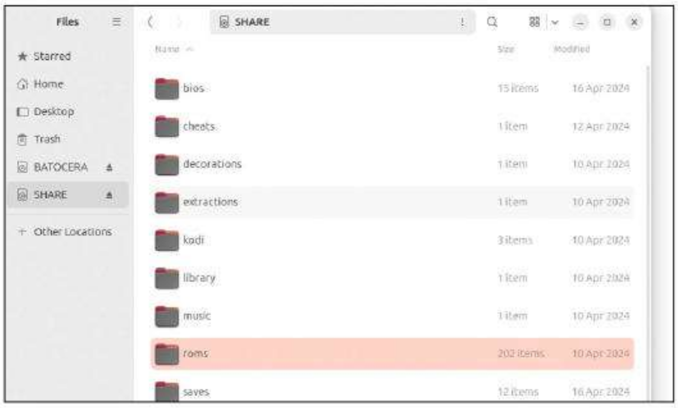
4 Use a microSD card

Short of USB adaptors or a dock? You can also copy ROM and BIOS files directly to the microSD card from your PC. This only works on Linux (you might have luck with WSL on Windows). Open your file browser from the terminal using *sudo* to ensure the correct permissions.



5 Copy ROMs to microSD

With the Batocera microSD card open on your PC, find your ROM files and copy them to the corresponding directory. So, NES games go in the **roms/nes** directory, for example. If you have used *sudo* to open the file browser, this should happen without a hitch.


Files
Starred
Home
Desktop
Trash
BATOCERA
SHARE
Other Locations
SHARE
Name
Size
Modified
bios 15 items 16 Apr 2024
cheats 1 item 12 Apr 2024
decorations 1 item 10 Apr 2024
extractions 1 item 10 Apr 2024
kodi 3 items 10 Apr 2024
library 1 item 10 Apr 2024
music 1 item 10 Apr 2024
roms 202 items 10 Apr 2024
saves 12 items 16 Apr 2024

6 Eject the microSD card

When copying files to the Batocera microSD card is complete, close any open windows and files, and safely eject the card. Return the microSD card to your Steam Deck and reboot the console to find the added ROMs. Whenever you add ROMs, restart to refresh the library.

discovery mode. A moment later, the controller should be discovered and added automatically. Alternatively, you may need to select the controller from a list of devices. Tap B (or the equivalent) on the controller to exit the menu and ensure it is working as intended.

It's a good idea to have a mouse and keyboard handy too, not least to help with adding retro ROMs.

Add ROMs to Batocera

Different options are available for copying ROMs and BIOS files *(see boxout below-right)* from your collection to Batocera on the Steam Deck.

You may prefer to copy them to a USB stick, then attach that to your Steam Deck via a USB hub, the dock or a USB-C to USB-A adaptor. Open the file manager using F1 on a keyboard or the Steam Deck's ellipsis button, then copy the ROM files from the USB stick to the Steam Deck. ROM files should be placed in the appropriate subdirectory of the **roms/** folder. Note that in each **roms/** folder there is a file called **_info.txt**, which lists the matching ROM file extensions.

While USB sticks are simple enough, it isn't the most convenient way to move data to a Steam Deck.

Of course, you also have the option of shutting down the Steam Deck, ejecting the microSD card, and adding ROM files that way. This doesn't require any messing about with cables and USB hubs, so is much quicker. By default, you probably can't copy your ROMs to the microSD card due to permissions. An easy way to fix this is to open a terminal and enter:

```
$ sudo nautilus
```

This launches the file browser with elevated permissions (change the file browser name to the one you have installed on your distro). You can then copy ROM and BIOS files to the corresponding directories on the Batocera microSD card.

Alternatives to Batocera

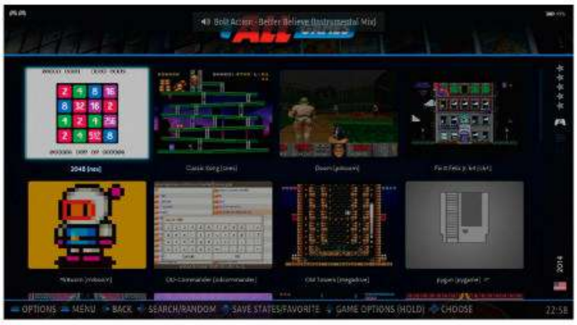
Batocera isn't the only game in town for retro gaming on a Steam Deck. While this solution neatly separates the classic gaming action from the main SteamOS experience, alternatives take a different approach.

The main competitor to Batocera is *EmuDeck*, an installer script that can be downloaded from **www.emudeck.com** to SteamOS. Once installed, you access emulators and ROMs via the RetroArch front-end.

EmuDeck's main benefit is that you don't need to sacrifice your Steam Deck's microSD card slot for dedicated retro-gaming solution. This is particularly useful for unmodified Steam Decks and 64GB models.

Meanwhile, *RetroDeck* (**http://retrodeck.net**) is a Flatpak that uses ES-DE (EmulationStation Desktop) to deliver retro gaming to the Steam Deck. Like *EmuDeck*, it installs within SteamOS, and has the added benefit of including all of your retro games within the Steam Deck's carousel. Despite it being considered as "in development", this seamless approach has resulted in *RetroDeck* becoming arguably the most popular retro gaming suite on Steam Deck.

Another option is RetroArch itself, which is actually available in the Steam Store (**https://store.steampowered.com/app/1118310/RetroArch/**). With this installed, you can manage ROMs and BIOS files on your Steam Deck; this version offers far better access to storage than RetroArch on some distros. If you would like to keep the retro-gaming experience purely within the Steam Deck's main user interface, use this.



Any classic game you can think of can run in Batocera. To get you started, open source games are included.

While plenty of retro games can be found within the Steam Store if you want to play it safe, a nice selection can be found on **GOG.com**. This is largely MS-DOS era PC games, but titles from other platforms can be found in this online store. GOG offers a game client and while a Linux version is not available, it can be installed on the Steam Deck in much the same way as Windows games run. (At one point, it could be installed with Epic Games using software called *Heroic*, or within the *Lutris* retro suite.)

Retro gaming options on the Steam Deck are good, and are as simple or as in-depth as you want.

When you're done with Batocera, press Start and select Quit > Shutdown. Once the console powers off, remove the microSD card, and replace the one you removed (if appropriate). Finally, press the power button to start the Steam Deck as usual. LXF

QUICK TIP

Ensure you shut down the Steam Deck before attempting to boot into Batocera. Simply restarting the console will not work. Also, after Batocera is enabled, restart your console again to enable the user partition to auto-expand. This ensures you have enough space to copy ROMs to the device.

» ROM AND BIOS FILES

For emulators to work correctly, BIOS files are required. As with game ROMs, there is some confusion about the legality of their use.

BIOS (Basic Input/Output System) files are quite complicated to explain. One way to look at them is as firmware that ensures the emulator runs the desired ROM. Not all emulators require BIOS files.

ROM (Read-Only Memory) files are images of cassettes, disks, cartridges, CDs or even chips that have been dumped to a data file.

In most cases, neither can be acquired legally. Both are covered by copyright, which means downloading them breaches that copyright. If you own the original hardware of the device you wish to emulate, dumping its BIOS might be considered fair use. Similarly, if you own the original version of the title you wish to emulate and create your own ROM file, you're once again within fair use provisions.

The difficulties arise when you go beyond. Whether you should download ROMs and BIOS files is an easy question to answer: no you should not. Going beyond those clear rules is entirely your decision.

It is rules like this that have resulted in retro suites like Batocera to start shipping with open source and shareware classics. This way, there is something to play immediately after installation. But don't forget, there is a growing market for new games produced for retro platforms, designed to run on original hardware and emulators alike.

» BOOT UP OLD STUFF EACH MONTH Subscribe now at http://bit.ly/LinuxFormat

BACK ISSUES » MISSED ONE?

ISSUE 315
June 2024

Product code:
LXFDB0315

In the magazine
Discover how to upgrade to Ubuntu 24.04 LTS and join us on a journey through the world's most popular distro's 20-year history. And don't miss our *Roundup* of hacker distros, an in-depth look at the impending Epochalypse, detailed tutorials on live-streaming audio via the terminal, setting up Gnome, editing images in *Krita,* and archiving C64 tapes, and plenty more besides.

ISSUE 314
May 2024

Product code:
LXFDB0314

In the magazine
Join us as we explore Linux's infiltration of Windows, and discover how you can also get Windows apps running in Linux as we uncork the latest *Wine* release. And let's not forget a *Roundup* of time trackers, tutorials on vulnerability audits, setting up a home ebook server, streaming games, and more, along with distro reviews, news, and packed Pi and *Adminsteria* sections.

ISSUE 313
April 2024

Product code:
LXFDB0313

In the magazine
Discover how to use the ultimate hacker's toolkit, staying out of trouble while doing so. And join us as we take the Puppy Linux developer's new distro for a run and explore its container features. Plus, we have a *Roundup* of retro-gaming distros, an in-depth look at filesystems, tutorials on adding plug-ins to our **LXF** shell and getting more from your VMs, plus news, reviews and oodles more.

ISSUE 312
March 2024

Product code:
LXFDB0312

In the magazine
Blast off into the future with a look at the five best next-gen distros, and discover whether Raspberry Pi or Orange Pi is the best SBC for you. Plus, learn how to rescue retro media, add NPCs to your own point-and-click adventure, emulate an analogue computer, and lots more. We've also squeezed in hardware and distro reviews, a password manager *Roundup*, news and more.

ISSUE 311
February 2024

Product code:
LXFDB0311

In the magazine
Stay safe online by sending net nasties to *Pi-hole,* and read about the rise and fall (and rise again) of Basic, and the roll-out of the rolling-release Rhino Linux distro. You can also learn how to strengthen your shell history, set up a home CCTV system, add an inventory to your point-and-click adventure, and much more. Plus, check out our packed reviews, news and Raspberry Pi sections.

ISSUE 310
January 2024

Product code:
LXFDB0310

In the magazine
Learn Linux with our in-depth guide to getting started with Elementary OS, or explore a more security-centric distro as we dive into Tails. Plus, find out how to perfect the art of OCR, finish your *WordPress* website, recreate the BBC Micro, and make your own point-and-click adventure. And no issue of *Linux Format* would be complete without news, reviews, Pi projects and much more besides.

To order, visit www.magazinesdirect.com

Select **Single Issues** from the tab menu, then select **Linux Format.**

Or call the back issues hotline on **0330 333 1113**
or **+44 (0)330 333 1113** for overseas orders.

Quote the product code shown above and have your credit or debit card details ready.

USA? EU? THE MOON*?

UK readers turn to p16

SUBSCRIBE!

Don't wait for the latest issue to reach your local store – subscribe today and let *Linux Format* fly straight to you.

LEARN KRITA Inside the Photoshop-besting art package
HOTTER EMAIL Turbo-charging the classic Thunderbird
SAVING TAPES How to securely back up your retro media
LINUX FORMAT
The #1 open source mag
HACKER SP
NEW RELEASE!
UPGRADE TO UBUNTU!
Learn how to use the most popular Linux distro ever! It's free, slick, secure and offers years of support updates
HAPPY BIRTHDAY
20 years of Ubuntu!
PLUS: HOW TO
Build a Pi Pico oscilloscope
Write Commodore 64 games in C
Use AI to create better office documents
FEAR Y2K 2.0! The year 2038 cometh and we're all doomed!
CLASSIC DEMOS Code classic Amiga demos in simple C
SWEET SOUNDS Make beautiful music with enhanced USB

2 GREAT WAYS TO SUBSCRIBE

Print + NEW digital access or digital-only!

USA	REST OF THE WORLD	EUROPE
From $135.49	From $135.49	From €119.49
For 13 issues	For 13 issues	For 13 issues

IT'S EASY TO SUBSCRIBE!

Visit **www.magazinesdirect.com/linux-format**

Call **+44 0330 333 1113**

Lines open Monday-Friday, 9am-5pm GMT

Or **1-844-779-2822**

Lines open Monday-Friday 8.30am-5pm EST

*We don't actually deliver to the Moon. Yet.

EMULATION

Credit: https://github.com/geo255/microtan65

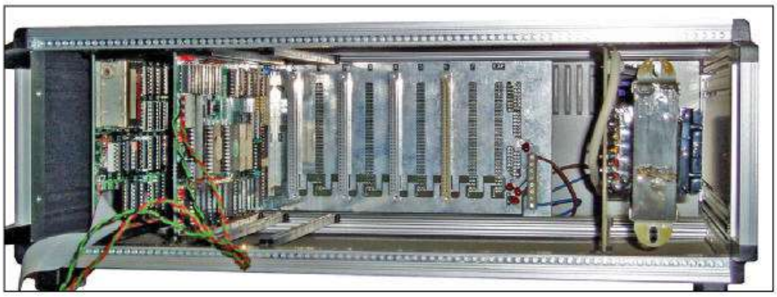
UK's first expandable home computer

The Microtan 65 was basic but expandable way beyond other '80s home computers. **Mike Bedford** explains how to experience it via emulation.

OUR EXPERT

Mike Bedford had a Microtan 65, bought in kit form, as his first ever computer. He eventually expanded it to run Basic, then underwent some soul searching to decide if he could justify the cost of upgrading it to 8K of RAM.

Although the Sinclair ZX80 is commonly thought of as the machine that kick-started the '80s home computer boom in the UK, an earlier machine may have also played a part, as we saw two issues ago. The Science of Cambridge MK14 (see **LXF314**) came three years earlier, was cheaper, and sold in significant numbers. But even the MK14 wasn't alone in these pre-ZX80 days, which brings us to our subject here.



This system contains just a Microtan 65 and a TANEX card, but the full-sized motherboard means it would have been well placed for future expansion.

CREDIT: Ian Dunster

In 1979, a now barely remembered company called Tangerine Computer Systems introduced the Microtan 65 home computer. As we'll see, it has similarities to both the earlier MK14 and the soon-to-be-released ZX80. And the good news is that you can get to grips with it through emulation.

Spotlight on Microtan

The Microtan 65 was similar to the MK14 in that it was supplied as just a single board with no power supply or case. Also like the MK14, it was expandable by the addition of expansion boards. As we'll see later, though, it was very much more expandable than the MK14, and this was one of its main selling points to the more serious enthusiast.

Turning to the ZX80, the Microtan was also similar to this machine in having a video display, rather than the MK14's onboard LEDs. And unlike the MK14, which was only available as a kit, the Microtan was, like the ZX80, available either as a kit or ready built. The price of the two computers weren't too dissimilar either, the Microtan 65 costing £79.93 in kit form.

In other ways, though, the Microtan 65 was quite different from the pioneering MK14 and the ZX80. For a start, the processor was neither the somewhat unusual SC/MP found in the MK14, nor the Zilog Z80

QUICK TIP

The Microtan 65 wasn't just a home computer, it appears it was also used in industry. One application we know of was prototyping and developing software for an arcade games console, even though the product eventually migrated to dedicated hardware.

» CHILDREN OF THE '70S

Microtan 65 wasn't alone in the late '70s in being a British home computer designed for enthusiasts with an electronics leaning. Perhaps its main competitor was the Nascom. Launched in 1977 for £197.50, it was also provided as a kit of parts, although fully assembled computers came later. In many ways it was similar to the Microtan 65, but it had a 1MHz Z80 processor, and 2K RAM. Perhaps the biggest advantage over the Microtan 65, though, was the provision of a QWERTY keyboard as standard. And, when the Nascom-2 was launched just a couple of years later, it had onboard Basic and the RAM capacity was boosted to 8K.

However, the Nascom-2 was different from its predecessor in a more fundamental way. Like the Microtan 65 without the addition of TANEX, the Nascom-1 didn't have buffering of its external signals. So, unless you bought the separate buffer board – and these were hard to get hold of – you couldn't expand it by adding extra cards. The Nascom-2 offered this capability, allowing it to become the heart of a fully expanded system. Expansion options were much the same as for a Microtan 65 system: high-resolution graphics, more RAM, more EPROM storage and a floppy disk interface. Oh, and there was also a Nascom-3 but that was just a fully assembled Nascom-2 in a box.

used in the ZX80. However, the Tangerine machine did go the same way as several of the ZX80's successors in using the 6502 processor. In many other respects, though, the Microtan 65 followed a very different path from virtually all the home computers of the early '80s. The MK14 had an onboard hex keypad while nearly all the later home computers had QWERTY keyboards. The Microtan 65 had neither as standard, and this is one of the first examples of its expandable nature and the choices this provided. So, a keypad was available, as was a full keyboard, the latter being extraordinarily expensive in the late '70s, so many users started with the keypad and later upgraded to a keyboard. In fact, with just the basic Microtan 65 board, the hex keypad was entirely adequate since it didn't have Basic, so programming was via machine code.

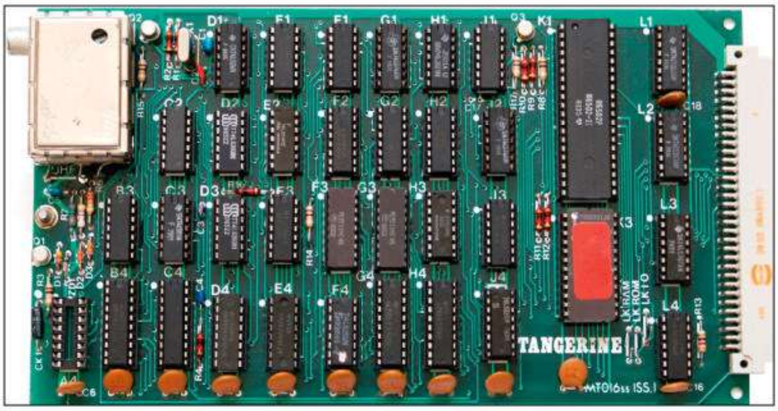
The basic Microtan 65 had a 6502 processor clocked at a modest 750KHz, 1K of RAM, which wasn't all accessible to users because part was used as video memory, and a 1K ROM that contained the *TANBUG* monitor program. The video circuitry generated 16 lines of 32 characters in upper case and monochrome. Onboard expansion options were available for lower-case letters and chunky graphics at 64x64 resolution.

The first major upgrade was via TANEX, another circuit board the same size as the Microtan 65. In fact, the boards were an industry standard size that used a 64-pin indirect edge connector. This differs from direct edge connectors – an extension of PCB tracks to the board edge – as found on PCI expansion cards for today's PCs. TANEX added 1K of RAM as standard, but it had sockets to allow a maximum of 7K, increasing the system capacity to 8K. There were also several ROM sockets that were mostly used for *XBUG* – an extension to the *TANBUG* monitor software that provided a primitive assembler and disassembler – and a Basic language interpreter. Neither of these were supplied as standard, though, so they had to be bought separately. TANEX also provided serial and parallel ports, plus circuitry for storing and retrieving software to and from a tape on an audio cassette recorder.

To use a TANEX with a Microtan 65, it was necessary to buy a motherboard to connect them together. Unlike PC motherboards, the Microtan equivalents were passive boards with little if any onboard electronics. There were two options: a two-socket board that could only be used to accommodate a Microtan 65 and a TANEX, or a 12-socket board for those intending to expand further. Many users didn't house a single Microtan 65 or a Microtan 65 and TANEX combination in a case, even though a case was available for two-board systems. However, in expanding beyond this, some means of providing mechanical support for a growing number of cards was desirable. This took the form of a 19-inch card frame that housed a 12-slot motherboard, a Microtan 65 and a TANEX, up to nine expansion cards, and a power supply. Tangerine offered such an enclosure or standard off-the-shelf card frames could be used.

We suspect many users never opted for the fancy Tangerine case, but its availability further hints at the Microtan 65's expansion philosophy.

CREDIT: Ian Dunster, Alan Maunder



Undoubtedly the Microtan 65 was basic, but from little acorns mighty oak trees grow.

Beyond the Basics

Adding TANEX, *XBUG* and Basic, plus an ASCII keyboard to the Microtan 65 provided something not too dissimilar from a ZX80. However, unless a low-cost non-Tangerine keyboard was available, this combination would have cost significantly more than the all-in-one home computers of the early '80s. But we're not comparing like with like, as the Tangerine solution was ideally suited for upgrading to what we'd be inclined, today, to refer to as a proper computer. However, this expansion didn't come cheap, even if we compare prices directly without taking inflation into account. So, in the following narrative, do bear in mind that there has been 534% inflation since 1979, so prices should be multiplied by 6.34 to obtain prices that would be meaningful today. So, for example, to a user in 1979, buying a £79.93 Microtan 65 would have felt like spending over £500 today. So, let's now look at some common options for further expansion.

First was the high-res graphics card, although it was only high resolution in comparison to the Microtan 65's onboard graphics. Its 256x256 resolution was also an improvement on most other home computers of the early '80s. However, the high-res graphics card was monochrome, so for a colour display, three were needed. What's more, the high-res graphics card only supported black and white, not greyscale, so using three of them only provided an eight-colour display. Given that a ready built card cost £80, although they cost somewhat less if you bought three, this wasn't a cheap option. Nor was it cheap to add more memory, which brings us to the next expansion opportunity. The TANRAM card increased the system memory capacity by as much as 40K, but it cost £109 fully populated.

However, it was upgrading to disk-based, capable of running a disk operating system, that really set the

QUICK TIP

We refer to EPROMs but since that's no longer an everyday term, here's the low-down. It stands for Erasable Programmable Read-Only Memory, it was user-programmable, and was an early form of flash memory. Locations could only be reprogrammed after the whole chip was erased, though, and that was achieved by shining ultraviolet light through its quartz window.

»

An EPROM contained the monitor program and Basic, and perhaps user code. However, to overwrite it, the chip had to be erased by shining ultraviolet through its window.

Microtan 65 apart from its contemporaries. This disk upgrade didn't offer a hard disk, though, but one or more floppy disks. If your experience of PCs goes back less than 20 years, you might not have encountered floppy disks. The bottom line, though, is that today's closest equivalent is probably flash drives, but floppy disks had a very much smaller capacity. And so we come to the Microtan 65 upgrade path in a bit more detail.

The first requirement was a disk controller, which took the form of an add-on card that provided an interface to floppy disk drives. This cost £110 ready built. It allowed floppy disk drives to be attached, thereby allowing 100KB removable floppy disks to be read from or written to. Drives cost £220, and you could attach up to four, should you have deep pockets. Finally, to get all this to work, you needed a disk operating system, which went by the name of TANDOS. You could buy this on EPROM, but that used up address space, or you could buy it on disk, in which case code would be loaded into RAM as and when it was needed. TANDOS on disk cost £40. If you've added all this up and multiplied the total by that magic figure of 6.34, you'll have discovered that it would have cost £2,346 in today's terms to add just a single floppy disk drive.

Even this wasn't the end of the journey, though. TANDOS enabled you to save and retrieve programs and data to and from disk much more quickly and reliably than from audio cassettes. But it wasn't an industry standard, so software for TANDOS wasn't exactly plentiful. What's more, this was an almost inevitable consequence of the Microtan 65's use of the 6502 processor, and this led to a dilemma. The Microtan 65 concept was one of non-redundant expandability – in other words, it could be upgraded without having to discard anything you'd bought previously. But if the 6502 took you to a dead end as far as a standard disk operating system was concerned, it had to go. The solution was to discard both the Microtan 65 and the TANEX and replace them with a single card. That single card combined the functionality of both these cards and more but, importantly, could take either a 6502 or a 6809 processor. And the 6809 processor paved the way to the Flex disk operating system.

Emulation time!

Written by Microtan 65 enthusiast Geoff MacDonald it was originally written for Windows, but has recently been ported to Linux. You won't find it in the repositories, so head off to **https://github.com/geo255/microtan65** and install it manually. It's a build-it-yourself job, but it worked fine for us. Geoff's GitHub page also has a good selection of sample programs – mostly games – for you to try. The **Readme** file provides information on the various games.

The emulator is of a fairly basic system comprising a Microtan 65 plus a TANEX with *XBUG*, Basic and the full 7K RAM. Although this is all you need to get a feel for what was probably the most popular Microtan 65 configuration, there are plans to emulate more advanced systems by adding three high-resolution graphics cards for a 256x256 eight-colour display.

To help you get a feel for the Microtan 65 through emulation, you'll find it useful to take a look at the original documentation, and you can find most of it at **www.geoff.org.uk/microtan/documentation.htm**.

When you fire up the emulator, the display starts with almost a full page of random characters – such was the imperative of saving the few bytes necessary to start with a blank screen – with meaningful information only at the bottom of the screen. That meaningful information is the single word "TANBUG", which indicates that the *TANBUG* monitor is running, and the striped block on the next line is a prompt for you to enter a command. And this begs the question of what commands you might like to try to get a feel for what it was like to own a Microtan 65.

TANBUG provided a handful of commands for viewing or editing memory locations, most commonly used for entering programs as machine code, and a means of running your code and debugging it. Since this was about the limit of what you could do with an unexpanded system, this is a good place to start. To see the contents of a memory location, we use the `M` command. That actually stands for Modify, which sounds paradoxical, but it'll soon become clear. To give an example, type `M0400` followed by Enter. This displays the value at 0400H, which is the first byte of user memory. *TANBUG* responds with something like `M0400,xx,`, where xx is the value in hexadecimal. If you now press Enter, the command terminates. However, you have the option to modify the memory location by entering a new value. Having entered that value, you can press Enter to terminate the command, Enter on the keypad (which corresponds to LF on the Microtan keyboard) to move on to the next location, or Esc to move to the previous one.

Beyond *TANBUG*, the TANEX extension can also translate 6502 assembler mnemonics to machine code, but only one instruction at a time. Even so, this is simpler and less error-prone than assembling by hand. This is achieved using the `T` instruction, which stands for Translate. There is also an `I` instruction, which stands for Interpret and which is a single instruction disassembler. If you want to give it a try, consult the TANEX manual for details of these instructions and the Quick Guide for a sample program.

Running Basic perhaps isn't too informative since the Microtan 65 Basic is a version of Microsoft Basic. And, while that means it's fully-featured, it's not too dissimilar from many other versions of Basic on '80s home computers. To start Basic, just enter `BAS` from the *TANBUG* prompt and press Enter. You're asked a couple of questions, which you can ignore by pressing Enter in response to each. Basic then announces that it's up and running, and at this point, you'll probably want to confirm that by entering and running some simple code. Once you've got Hello World out of your system, and listed out all the numbers from one to 10, though, what next?

We suggest you create some code to use the Microtan 65's block graphics. To learn how to do that, take a look at the Quick Guide and, in particular, at the sections entitled Display Format and Character Sets. And while this is enough for you to start writing some

QUICK TIP

Tangerine wasn't a single-product company; it followed up on the Microtan 65 with the Oric. Launched in 1983, it cost only slightly more than the Microtan 65 but had a much higher specification, and it came in a box that also contained a QWERTY keyboard. It was a 'me too' product, though, and didn't take the world by storm.

CREDIT: Raimond Spekking/CC BY-SA 4.0 https://commons.wikimedia.org/wiki/File:Eurotec_ET2764Q-5370.jpg

» DIY EXPANSION OPTIONS

The Raspberry Pi is considered a computer for electronics enthusiasts, but most owners restrict their tinkering to adding external circuitry via the GPIO (General Purpose Input Output) port. The potential for the enthusiast to customise their Microtan 65, however, was much more fundamental.

Adding hardware to an RPi is via a USB port, the dedicated camera or display ports, or via the GPIO port. Most experimenters customise their RPi through the GPIO, starting with interfacing LEDs and switches, and working up from there. However, while this allows an RPi to be connected to real-world hardware, it doesn't allow additions to the core processing capabilities. The Microtan 65 was different in this respect in providing the user with access to the microprocessor's fundamental signals, such as the address bus and the data bus.

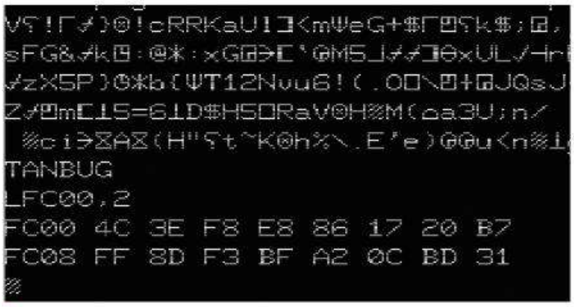
It would thus be possible to design boards for extra memory or high-res graphics, even though Tangerine offered these options, as did third parties. When Microtan 65 users did design and build their own expansion cards, though, it was usually to provide less commonly required options, or drop-offs that Tangerine hadn't thought of, such as real-time clocks, soundcards, EPROM programmers and EPROM emulators. And, as computing in the late '70s was considered an off-shoot of the hobby of electronics, these designs were often shared in hobby electronics magazines. It has to be admitted, though, and we speak from experience, it was a scary moment when you first switched on your pride and joy having just plugged in a homemade board.

chunky graphics to the screen in Basic, there's a shortcut. Do look at the Basic manual and, specifically, the Appendices. This provides a subroutine for setting a chunky pixel to black or white at any of the 64x64 positions on screen. Once you've got that running, how about wrapping some simple code around it, say to draw a diagonal line – so the chunkiness can be fully appreciated – from one corner of the screen to the opposite corner?

Now you've learned something of the Microtan's chunky graphics, and gained an appreciation of how this feature was coded, you might like to try some of the sample programs at **https://github.com/geo255/microtan65**. Many feature chunky graphics and, despite what we might think today, they kept enthusiasts enthralled back in 1979. However, the emulator doesn't currently allow programs to be loaded from within the emulator. Instead, they have to be loaded when you start the emulator, by typing the filename after the `./microtan65`, the sample programs being the files with an M65 extension. Apparently, loading programs from within the emulator is a feature that's planned for a future release.

Getting physical

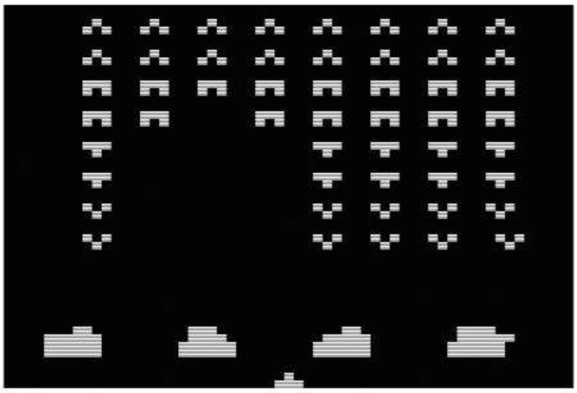
This isn't for everyone, admittedly, but if you have a leaning towards the hardware side of computing as well as the software, it could be of interest. After all, many of the original users of the Microtan 65 had a passion for electronics. This doesn't mean you need to design circuit boards from scratch, though. That job has been done by Alan Maunder, who used a Microtan 65 in the '80s and hankered to relive that experience, having got rid of his machine some time back.



```
V⸮!Γ≠}◎!cRRKaU!]<mΨeG+$Γ⸮⸮k$;⊡,
sFG&≠k⊟:@✱:xG⊡>Ε'@M5⅃≠≠]θxUL/⊣r
≠zX5P}⊚✱b{ΨT12Nvu6!(.O□\⊟+⊡JQsJ
Z≠⊟mΕ⊥5=6⊥D$H5□RaV⊚H▨M(⌂a3U;n/
 ▨ci>⧖A⧖(H"⸮t~K⊚h%\.E'e)@@u<n▨⊥
TANBUG
LFC00,2
FC00 4C 3E F8 E8 86 17 20 B7
FC08 FF 8D F3 BF A2 0C BD 31
▨
```

A basic Microtan 65 was limited to little more than viewing or editing memory locations, and running and debugging machine code.

If you head to his website at **www.microtan.ukpc.net**, in particular to the Shop, you'll find several PCBs. He also sells EPROMs programmed with *TANBUG*, TANEX and so on, having received permission to distribute the code from Dr Paul Johnson, co-founder of Tangerine. Adding up a few prices, it costs £46 to buy the PCBs and EPROMs to build a two-board system comprising Microtan, TANEX, two-board mini-motherboard and keypad. Separately you'd need to buy the components, and the news on availability is good. The resistors, capacitors and small chips are all widely available, and there's even a currently manufactured version of the 6502 processor, the 65C02. Regarding the other chips, Alan says, "Some of the components such as the 2114 and 6551 can be difficult to source, but they do crop up on eBay from time to time. I always ask my customers to let me know if they have difficulty finding components. To date I've not had any showstoppers." LXF



Space Invaders illustrates Microtan 65 graphics in all their blocky glory. And despite its primitive appearance, it truly was addictive.

» IMPROVE YOUR LINUX SKILLS Subscribe now at http://bit.ly/LinuxFormat

VIDEO CHAT

Make secure video calls with Linux

Well-known chatterbox **Nate Drake** walks newcomers to Linux through the many excellent videoconferencing options on your Linux system.

OUR EXPERT

Nate Drake is a tech journalist specialising in cybersecurity. He placed his first Linux video call via *Skype* in 2007 to ask his girlfriend to buy him some Smarties.

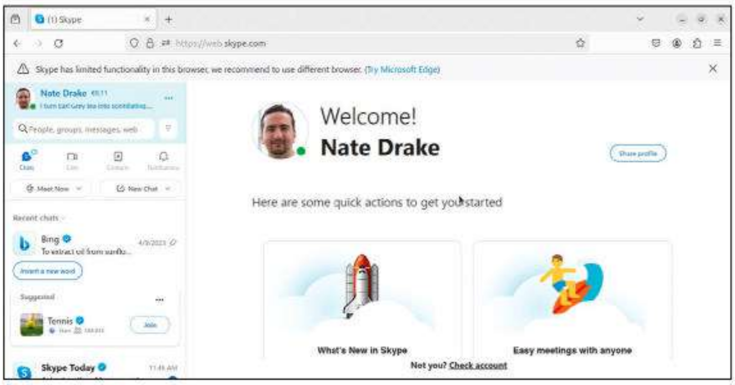
QUICK TIP

If you're installing certain software like the UVC driver in Linux, you need to download packages specific to your kernel version. If you're unsure which you have, simply open the terminal and run `uname -r`.

If you're new to Linux it's easy to fall for the stories peddled online that the OS is terrible for more advanced activities such as gaming and videoconferencing and you should stick to Windows or Mac OS.

Still, the Linux desktop's market share has climbed to over 4% in 2024 and it seems developers are paying attention. A number of popular video-calling applications, such as *Skype, Zoom* and *Signal,* offer Linux versions. If you're conscientious about privacy, there are also open source, encrypted video-calling options like *Jitsi.*

In this guide, we explore how to get started with videoconferencing in Linux, as well as how to set up some of the more popular video-calling apps.



Skype has limited functionality in this browser, we recommend to use different browser. (Try Microsoft Edge)

Welcome!
Nate Drake

Here are some quick actions to get you started

The Skype web interface (pictured here running in Firefox) is optimised for running in Chromium-based browsers. We recommend using Chromium itself.

Testing your webcam

Webcam support in Linux is mostly provided by the UVC driver. This is designed to make sure webcams 'just work' in the same way as when you connect different types of USB drives.

UVC compliant peripherals like webcams only need this driver to work out of the box. In an ideal world, this should be the case for your integrated/USB webcam.

The best way to test this is to use a video camera app. If you're using the most recent version of Ubuntu LTS, the traditional program *Cheese* has been replaced with the Gnome app *Snapshot.* You can reinstall *Cheese,* however, by opening a terminal and running:

```
$ sudo apt-get update
$ sudo apt-get install cheese
```

If you don't feel comfortable with the command line just yet, you can also search for *Cheese* in the *App Center,* then hit Install to begin setup.

Once the app launches, you should see a live feed of your own face staring back at you. If this is not the case, hit the three lines at the top-right and choose Preferences. From here, choose the Webcam tab and find the Device field. You can use the drop-down menu to check your webcam has been correctly selected.

If your webcam isn't recognised, first double-check that the issue isn't related to *Cheese* by trying an alternative app. Search the *App Center* for *Guvcview,* which, like *Cheese,* can view and record video from UVC-compatible devices. Click Install to download. Alternatively, fire up a terminal once again and run:

```
$ sudo apt-get guvcview
```

Launch the app to try to detect your video device.

Webcam woes

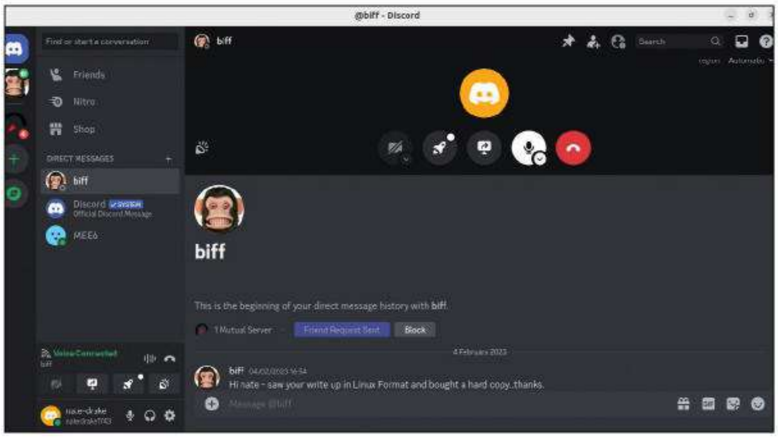
Assuming your chosen webcam is not recognised by Linux, you have two choices. The first involves downloading the source code for the UVC driver for your version of the Linux kernel, then modifying it to work with your webcam. This is an involved process and doesn't always work. You can find further help with this in the community support forums for your distro.

Or you can invest in a UVC-compatible webcam. These days retailers are aware of Linux users, so you can search sites like Amazon to find a Linux webcam.

Check if Linux is listed under supported operating systems. Ideally, the specs should also confirm it's UVC-compatible. For instance, on **Amazon.com** (other huge megacorps are available) we found the Septekon 1080p Webcam, the specs of which specifically state it supports UVC mode for Linux 2.6.26 or above.

Skype setup

Assuming you now have a working camera, it's time to put it to good use. *Skype* was first released over 20 years ago, though its Linux client has only been in


@biff - Discord

This is the beginning of your direct message history with biff.

4 February 2023

biff 04/02/2023 16:54
Hi nate - saw your write up in Linux Format and bought a hard copy..thanks.

Discord supports video calling on Linux through your browser or the app (pictured). The website maintains performance is better with the client.

active development since 2016. *Skype* is one of the best ways to hit the ground running in Linux, as you don't need to install the client to place video calls. Just fire up a browser and head to **https://web.skype.com.**

From here you can sign in to your *Skype* account and place calls right away. When you place a call for the first time, your browser may ask permission to access your webcam and/or microphone. The web interface is optimised for *Chromium*-based browsers, so you may see an alert if you use alternatives like *Firefox.*

Although there's a Linux version of *Chrome* available, we recommend using the open source *Chromium* itself. On Ubuntu, you can install this via the *App Center.* Alternatively, open a terminal and run:

```
$ sudo apt-get install chromium-browser
```

If you're not using Ubuntu or simply prefer to avoid Snap packages, you can also download the latest version of *Skype* for Debian-based distros by running:

```
$ wget https://repo.skype.com/latest/
skypeforlinux-64.deb
```

You can then install *Skype* by running:

```
$ sudo apt install ./skypeforlinux-64.deb
```

However you install the *Skype* client, it automatically launches each time you log in to the Linux desktop. To disable this in Ubuntu, click the Activities tab and search for Startup to open Startup Applications Preferences. Highlight Skype and choose Remove.

Zero in on Zoom

Although *Zoom* has been around since 2012, it was during the pandemic that it became the de facto app for videoconferencing, being downloaded 500 million times during 2020 alone.

Most importantly for users, there's a version of *Zoom* for a number of the most popular Linux distros, including Ubuntu, Debian, Mint, Red Hat and Fedora.

To get started, simply open your browser and go to **https://zoom.us/download?os=linux**. Click the drop-down menu marked Please Select A Linux Type to select your OS. Now click the Download button.

Once the package download is complete, if you're using a Debian-based system, you can install via the terminal. First switch to the **Downloads** folder using:

```
$ cd Downloads
```

Next, install the package by running:

```
$ sudo apt install ./zoom_amd64.deb
```

Enter `Y` then press Return to continue. Once install is complete, you can launch *Zoom* via Activities.

We found numerous complaints online about *Zoom's* Linux implementation, mostly focused around choppy audio and screen sharing. If you encounter issues like these, you may prefer to use the *Zoom* web client. If you've been invited to a *Zoom* meeting, you can do this simply by clicking the link to join. A pop-up window may appear prompting you to install the *Zoom* client for Linux – just click Cancel if this happens. Next, scroll to the bottom and choose Join From Browser.

If you're the one hosting a *Zoom* meeting, simply go to **https://zoom.us** in your browser. Click Sign In at the top-right and enter your password. Once done, you can

» WHAT'S UP WITH WHATSAPP?

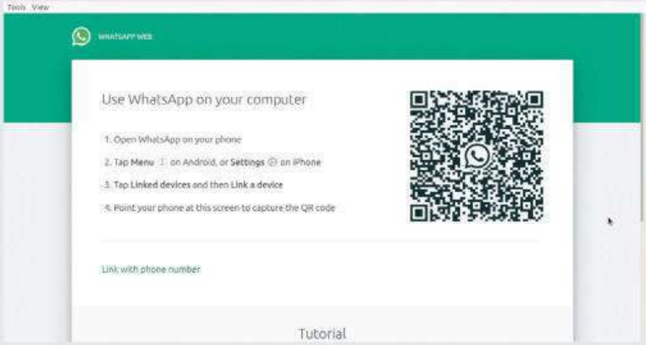
You may have noticed that we haven't yet mentioned *WhatsApp*. Don't worry, we haven't overlooked the messaging app of choice for three billion people.

If you simply want to exchange messages and voice clips with fellow *WhatsApp* users on Linux, you're in luck. Just point your browser to **https://web.whatsapp.com**. From here you can use your *WhatsApp* mobile app to scan the QR code by going to Settings > Linked Devices. The *WhatsApp* interface then loads, minus any audio/video calling options.

The issue is confused further by the existence of third-party *WhatsApp* clients. If you're an Ubuntu user, simply running a search for 'WhatsApp' in the *App Center* shows nine different supposed *WhatsApp* clients.

Many of these, such as *WhatsDesk*, are legitimate but ultimately are just an electron wraparound for the web app, combined with desktop notifications. In other words, such Linux apps offer you very little compared to just loading the website in your browser.

Despite not supporting video calls, the *WhatsApp* web app alerts you to incoming calls, though you see a prompt stating: 'Check your phone to answer'.


Use WhatsApp on your computer

1. Open WhatsApp on your phone
2. Tap Menu ⋮ on Android, or Settings ⚙ on iPhone
3. Tap Linked devices and then Link a device
4. Point your phone at this screen to capture the QR code

Third-party apps such as WhatsDesk enable you to use WhatsApp web as a standalone client but don't offer extra functions like video calling.

Meta has yet to announce any plans to develop a dedicated Linux *WhatsApp* client. The Arch Linux wiki also claims that the company has "strictly attempted" to prevent third-party clients from using *WhatsApp's* protocol. Technically, users could install the *WhatsApp* Android app in an emulator like *Anbox* running on Linux, but this is unlikely to result in good video call quality.

»

QUICK TIP

Ubuntu's online documentation has a dedicated help page for troubleshooting webcams (https://help.ubuntu.com/community/Webcam/Troubleshooting). This can help you resolve issues such as low brightness as well as upside-down or mirrored images.

choose Web App at the top-right of the screen and pick an appropriate option – Meetings, for example.

Gabbing with Google Meet

Google Meet came out in 2017 and has served as successor for both Hangouts and *Duo*. Like *Zoom*, its usage spiked during the pandemic, with around 200 million users a day in April 2020.

Part of its popularity lies in the fact that *Google Meet* is designed to be truly cross-platform. In order to get started, just point your browser to **https://meet.google.com** and click New Meeting to begin. .

Although we didn't observe any noticeable difference during our tests, Google's dedicated page insists that you can "find your meeting window faster and easier" with its PWA (Progressive Web App).

This effectively allows you to install *Google Meet* as a standalone app in Linux. To proceed, you need to be using a *Chromium*-based browser. Once this is installed, head to **https://meet.google.com** once again.

Once the page loads, you'll see a small icon in the address bar that you can use to install the PWA. Once installed, you can search for and launch *Google Meet* in the same way as any other app in your Linux OS.

Soliloquise in Signal

Signal Messenger is an excellent free and open source app. Like *WhatsApp*, it uses the Signal Protocol to secure conversations using end-to-end encryption, making it popular with people who care about privacy.

While it doesn't have as many bells and whistles as *WhatsApp*, the platform does support video calling, and a Linux client has been in active development since 2017. You need to install this on your Linux device because, unlike *WhatsApp*, there's no web app.

Ubuntu users who are keen to get started can simply search for *Signal* via the *App Center* and click to install the Snap version. Alternatively, open your Linux web browser and head to **https://signal.org/download/**. Click Download For Linux to view the instructions for installing via the terminal. At the time of writing, these are only for 64-bit Debian-based Linux distros.

First install the *Signal* software public signing key:

```
$ wget -O- https://updates.signal.org/desktop/apt/keys.asc | gpg --dearmor > signal-desktop-keyring.gpg
$ cat signal-desktop-keyring.gpg | sudo tee /usr/share/keyrings/signal-desktop-keyring.gpg > /dev/null
```

Next, add the official *Signal* software repository:

```
$ echo 'deb [arch=amd64 signed-by=/usr/share/keyrings/signal-desktop-keyring.gpg] https://updates.signal.org/desktop/apt xenial main' |\
 sudo tee /etc/apt/sources.list.d/signal-xenial.list
```

Now update your system and install *Signal* with:

```
$ sudo apt update
$ sudo apt install signal-desktop
```

On first launch, as with *WhatsApp*, *Signal* asks you to scan a QR code using your mobile version of the app by going to Settings > Linked Devices > Link New Device. The first time you place a call, *Signal* requests access to your webcam and microphone.

Discord diatribes

Although *Discord* is a popular VOIP platform for gamers, it's home to thousands of communities on a diverse range of topics. The simplest way to get started with *Discord* in Linux is just to use your browser to head to **https://discord.com/**. From here you can click Open Discord In Your Browser to enjoy all the platform's features, including video calling.

If you do, you'll notice an intermittent banner at the top that states that for "better performance and an in-game overlay" users must download the desktop app.

The app is Electron-based, so it simply serves as a wraparound for the web app. If you want to go ahead,

» ALL ABOUT AUDIO

While webcams do have an integrated microphone, these aren't always good quality. You may also hear yourself on the call, which can be distracting, as well as distortion and feedback.

The easiest solution is to use a dedicated headset when on calls. Wired Logitech USB models start from as little as £29 on Amazon.

Most of these models plug and play out of the box, but always check the manufacturer's product specs carefully to be sure it's Linux compatible.

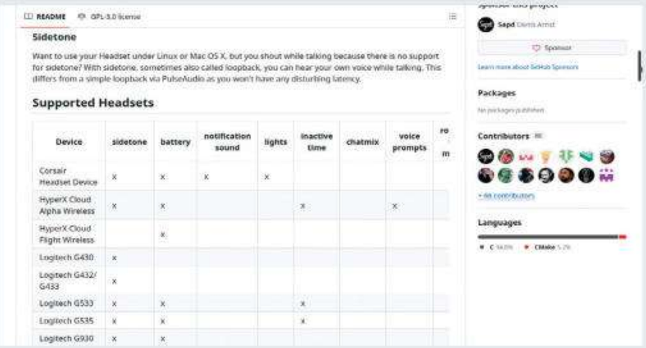
Many manufacturers, such as Logitech, include Windows software with headsets to control aspects such as sidetone, LEDs and inactivity periods. Depending on your headset model, you may be able to use *HeadsetControl* (**https://github.com/Sapd/HeadsetControl**) instead.

This is a command-line utility (though a shell extension GUI exists for Gnome) that can fine-tune your headset's audio settings. If you have a compatible model, you can download and compile *HeadsetControl* via the terminal.

First, make sure you have the required packages to do this by running:

```
$ sudo apt-get install build-essential git cmake libhidapi-dev
```

Next, copy the source code to your device and switch to the new folder:



Sidetone

Want to use your Headset under Linux or Mac OS X, but you shout while talking because there is no support for sidetone? With sidetone, sometimes also called loopback, you can hear your own voice while talking. This differs from a simple loopback via PulseAudio as you won't have any disturbing latency.

Supported Headsets

Device	sidetone	battery	notification sound	lights	inactive time	chatmix	voice prompts	ro m
Corsair Headset Device	x	x	x	x				
HyperX Cloud Alpha Wireless	x	x			x		x	
HyperX Cloud Flight Wireless		x						
Logitech G430	x							
Logitech G432/ G433	x							
Logitech G533	x	x			x			
Logitech G535	x	x			x			
Logitech G930	x	x						

The HeadsetControl GitHub page maintains a full list of compatible headsets as well as which specific features are supported.

```
$ git clone https://github.com/Sapd/HeadsetControl && cd HeadsetControl
```

Now you can create a dedicated directory to start building the code:

```
$ mkdir build && cd build && cmake ../
```

You can now install the tool via these two commands:

```
$ make -j4
$ sudo make install
```

Once installed, you can view all available commands for your audio headset via:

```
$ headsetcontrol -h
```

you can find a Snap version in Ubuntu's *App Center.* There's also a Flatpak available via Flathub, which Linux Mint users can install via the *Software Manager.*

If you're running a Debian-based distro, you can also install the package by downloading directly from the main website. Use your browser to visit **https://discord.com** once again and click Download For Linux. Choose Deb to start the download itself. Once this is complete, open a terminal and switch to your **downloads** folder:

```
$ cd Downloads
```

From here, set up the relevant package; for example:

```
$ sudo apt install ./discord-0.0.48.deb
```

Upon launch, the app automatically searches for and applies any pending updates.

Talk on Telegram

Telegram is relatively new, released in 2013. Still it currently claims over 900 million active monthly users, due no doubt in part to its open source Qt-based desktop client, which is Linux compatible.

Like *Discord*, there are both Snap and Flatpak versions available meaning you can install either via Ubuntu's *App Center* or via Flathub.

If you visit **https://desktop.telegram.org**, you can also download the client directly by clicking Get Telegram For Linux x64. The necessary files are compressed in a tarball. To extract them, simply open the **Downloads** folder, right-click on the relevant **tsetup** file and choose Extract Here. Once this is done, open the **Telegram** folder and double-click on the executable of the same name to launch.

As the client is open source, there are also some Linux-friendly forks including *Kotatogram* (**https://kotatogram.github.io**) and *64Gram* (**https://github.com/TDesktop-x64/tdesktop/releases**).

The platform also has a web portal, which can be accessed at **https://web.telegram.org/** The portal supports making video calls and, unlike other messaging platforms we found, doesn't claim that its desktop client offers a better experience.

Jawing on Jitsi Meet

While the clients used to access *Telegram* are open source, the server-side software is proprietary. This makes it difficult to be sure how much your conversations and video calls are being monitored and which (if any) of your personal data is being stored.

The Jitsi Project offers a solution via a comprehensive collection of free and open source multiplatform videoconferencing and messaging apps.

On the server side, if you're running a Debian-based distro, such as Ubuntu, you can visit **https://jitsi.org/downloads/** to download the necessary software. Jitsi maintains a dedicated guide for help with self-hosting at **https://jitsi.github.io/handbook/docs/devops-guide**. For best performance, use the Stable builds.

Jitsi Meet relies on WebRTC (Web Real Time Communication), which supports end-to-end encryption, making it one of the most secure options for videoconferencing on Linux.

Although there is a legacy desktop client, when we tried to install it in Ubuntu 24.04, setup failed due to unmet dependencies. Upon checking, we found the necessary packages had been deprecated.

On the plus side, since every modern web browser supports WebRTC, you can join or host a video meeting by visiting **https://meet.jit.si/**.

From here you can set a meeting name or use one assigned by the site at random, then click Start Meeting. You're prompted to allow microphone/camera access, as well as enter your name. Unlike other video-calling platforms, there's no requirement to register an account unless you want to be a meeting moderator.

Having praised *Jitsi's* fully open source nature and better overall security relative to other solutions, if you're using a VPN, WebRTC can be risky. This is because there's a danger of WebRTC leak, which can reveal your IP address even when connected to a VPN.

You can mitigate this to some extent by disabling WebRTC when not in use or configuring it via a special extension, such as Google's WebRTC Network Limiter.

If using a *Chromium*-based browser, each time you access *Jitsi Meet*, you're prompted to add an extension to integrate the platform with *Google Calendar* and *Office365.* This isn't compulsory but can be useful.

Video vagaries

Live video calls are one of the most resource-intensive activities, which can lead to glitches and errors. The most common complaint is delay between participants speaking and/or the audio being out of sync with video.

You can often correct this from within your videoconferencing app by reducing the webcam video quality – for example, from 1080p to 720p. Although many webcams have an integrated microphone, for optimum video calls you should try to use an external headset *(see box, opposite page).*

Individual videoconferencing applications can also have compatibility issues with various Linux kernels and drivers. If you're just starting your journey with Linux, we recommend a beginner-friendly distro such as Ubuntu, which has excellent online documentation.

If you're making video calls via a browser, also check the support pages of your chosen platform to find which works best. During our research for this article, most providers seem to recommend *Google Chrome.*

The open source *Chromium* browser on which *Chrome* is based should provide equivalent functionality. If, however, your provider insists that you need the real deal, you can install it via the terminal on Debian-based systems such as Ubuntu.

Download the latest *Chrome* package by running:

```
$ wget https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb
```

Next, install the package by running:

```
$ sudo apt install ./google-chrome-stable_current_amd64.deb
```

If you're using a desktop client and encounter video-calling issues, first double-check the provider's website and make sure you're running the most up-to-date version of the software. If video quality is still poor, ensure this isn't caused by a slow connection speed by siting your device closer to the router, or use a wired Ethernet connection to stop wireless interference. LXF

QUICK TIP

If you have *VLC Media Player* installed on Linux, you can also use it to test your webcam. Just launch *VLC*, then from the menu choose Media > Open Capture Device. Double-check your chosen camera is selected in the Capture Mode drop-down menu, then click Play.

» IMPROVE YOUR LINUX SKILLS Subscribe now at http://bit.ly/LinuxFormat

KRITA

Credit: www.krita.org

Part Two! Don't miss next issue, subscribe on page 16!

Feel the pressure with drawing tablets

After ignoring a key feature of Krita, under pressure **Neil Mohr** brushes away all criticism and paints over his mistakes.

OUR EXPERT

Neil Mohr hasn't got an artistic bone in his body and finds copying is very flattering.

Last issue, we took you on a whirlwind tour of *Krita,* largely ignoring the fact that it's primarily designed for digital artists and concentrating instead on getting used to the interface and looking at how it can be used for basic photo editing; selecting areas in various ways, manipulating colour and brightness, and touching up areas with the clone brush.

With the basic interface introduction out of the way, we can now spend time taking a closer look at *Krita's* powerful Brush Engine. It enables *Krita* to mimic a host of natural drawing media and allows users to customise brushes to their needs. While this all works fine with a mouse, it takes full advantage of pressure-sensitive input, so to get the most out of the Brush Engine, a graphics tablet is the best tool to use.

The good news is that there is a wide range of affordable and highly capable graphics tablets available for Linux. We're using the Wacom One Medium model that can be picked up for around £50 – though there's plenty on eBay for half this. The Small model costs around £30, though we'd find that size a little tricky to use. Linux is also ready to rock when it comes to most tablets – see the boxout on settings *(right),* but the main takeaway is you can just plug in a Wacom tablet and it'll work with tablet settings built into Ubuntu.

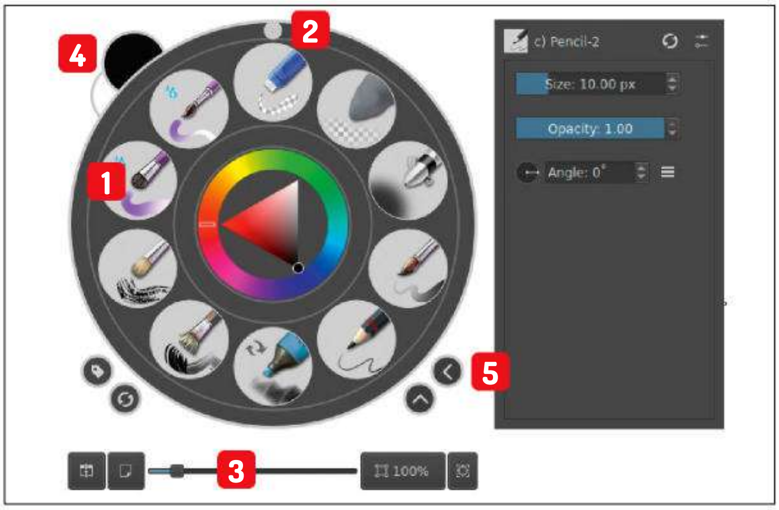
If you haven't done so already, fire up *Krita.* Before delving into the Brush Engine and settings, we'll point out the Pop-Up Palette, which gives you near instant access with a right-click to your favourite brushes, zoom levels, canvas rotation and colour settings. While it does provide fast access to brushes and settings, you might spot it's doubling up on settings available from the top toolbar (brush settings) and the Brush Docker (right dock strip). However, these disappear when working in its distraction-free full-screen mode and when the pop-up palette comes into its own, providing instant brush controls while you draw.

We should point out that you can change the available visible brush settings from the flip-out Brush HUD dialog. Click the top-right settings button and you'll find a list of available properties. The ones on the left are hidden while the right ones are visible, so you can add and remove them from the flip-out HUD.

The best way to get started, as with everything, is to dive in and try out these brushes. *Krita* offers a wide range of brush presets tailored for different artistic styles and tasks. You can access these presets from the Pop-Up Palette, as mentioned, or more fully in the Brush Presets Docker, bottom-right. Keep in mind that how you draw depends on the tablet you have. Outside of using a generic phone or tablet screen, most basic drawing tablets should provide at least pressure and speed sensing. More advanced models can enable *Krita* to modify drawing styles based on the tilt of the pen and rotation, as you would use a real paint brush.

To access the Brush Editor, with a brush selected press F5. This opens a complex settings window that lets you adjust the existing brush and also create entire

POP-UP PALETTE



c) Pencil-2

Size: 10.00 px

Opacity: 1.00

Angle: 0°

100%

1 Brush selection Your first 10 'favourite' brushes are accessible – you need to add and remove brushes from the favourite tag to manage these, though the bottom-left Tag button provides deeper brush access.

2 Canvas rotation Grab this little dot and you can quickly rotate the entire canvas to any degree.

3 Quick zoom controls The slider provides instant zoom control. The buttons either side provide mirroring, full-screen editing, 100% zoom and fit to window.

4 Foreground & background Displays the current foreground and background colours, editable from the central colour selector. Use X to toggle between them.

5 Brush controls Use the arrow button to toggle this window that provides further brush control, though many are available from the toolbar.

new types of brushes from scratch. You can adjust a phenomenal number of parameters, such as size, opacity, flow, spacing and shape dynamics. The window can be split into six main areas:

- **Left Preset Chooser** – this offers you a starting point from any of the existing brushes loaded into *Krita.*
- **Right Scratchpad** – use this blank canvas to quickly test out your brush settings and, when done, create a preview icon.
- **Top Information** – provides a preview and basic information on the current brush selected.
- **Bottom Options** – miscellaneous controls for handling erasing modes and temporary settings.
- **Options List** – the main selection of possible attributes you can apply to a brush.
- **Central Option Configuration** – for any of the selected main options, this is where you configure its exact settings.

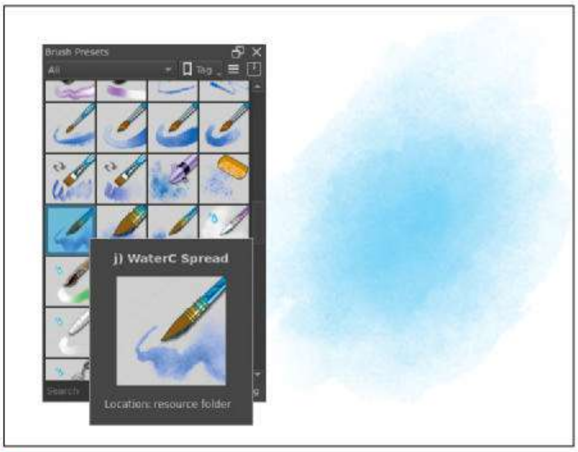
When creating a brush in *Krita,* they're made up of four basic parameters. The first is the brush tip, or shape. This can be from a mask image or procedural tip called an Auto brush. You apply various options that affect the brush, such as size, colour, source input, patterns and so on. Sensors are also applied – for us, that largely means pressure and speed adjustments, but aspects such as direction, distance, time, fade and more can play a part. Finally, each sensor has its own control schema for pressure that's based on a curve and effectively controls how heavily it's applied. These are the essential parts, but there are more optional controls on top of these, such as masks and textures.

Creating brushes

Let's have a play with the Brush Engine and create a brush from scratch, using it as an excuse to get used to using the tablet and exploring some of the *Krita* brush settings. To start, it's best to have black as the foreground colour, so press `d` (to set *Krita* to the default foreground/background colours of black and white). You can press `x` to swap the foreground and background colours. Press F5 to open the Brush Editor.

Open the left-side Preset box, if it isn't open already, by clicking the top-left arrow button. It's worth having a look through the many, many presets. At the top, you can filter by Engine and set Tags; alongside this, you can create Tags, too. At the bottom, you can search on names, too – type `Favourite` and it shows you those preset Favourite brushes that appear in the Pop-Up Palette. On the right, open the Scratchpad and test out the brushes; it draws in the selected foreground colour, hence ensuring we have black set. If nothing seems to happen, you can try the Gradient button at the bottom of this section to fill it with a greyscale. The other buttons here fill it with a preset icon, the current image or the background image, or erase it.

To start from new, click the bottom-left + button. This opens a list of base Brush Engines to base the new brush on. Select Pixel for this experiment. Our first task is to make the size of the brush vary with pressure applied. First, under Brush Tip, set the diameter to 80; this is larger than you'd probably want but it highlights the effect. Now select Size > Pressure (it should be on already) and test out the pen. You can also deselect



One of the preset brushes apes a decent water colour effect.

A drawing tablet can help unleash your creativity by offering a more natural drawing style.

QUICK TIP

We previously looked at the amazing G'MIC filters and effects within *GIMP* (see LXF288, LXF294) – as they're open source, they're also all available in *Krita*, so it's all transferable knowledge.

» TABLET SETTINGS

Wacom tablets have been supported by Ubuntu for almost its entire existence, going back to 6.04 Dapper Drake, as Wacom has been contributing to the kernel since 2002. So, when it comes to Wacom devices, you should find they work out of the box. If you head over to Settings > Wacom Tablet, you'll find two areas: Stylus and Tablet. Under Stylus, the main point of interest is Tip Pressure Feel. We set this quite low – Soft. It's the amount of physical pressure you need to apply for the tablet to respond and we find Firm really does need a firm press. You can also define behaviour for the two built-in buttons, but remember a pen tap is counted as a left mouse button press.

Under the Tablet tab, we personally prefer the Tracking Mode set to Touchpad (Relative) rather than the Tablet (Absolute) mode. With relative tracking, the mouse point stays where you left it, no matter where you next bring the stylus down on the graphics tablet. Some tablets have displays built into them, in which case Absolute would make more sense, as it also would when you get more used to using one. Finally, use the Map To Monitor option to handle multiple-monitor situations. The software enables you to map the tablet to just a single monitor, which can greatly help your workflow.

»

QUICK TIP

***Krita* users love creating brushes, so you'll find a host of questions and tips on doing just that on the official *Krita* forums, along with tons of other tutorials: http://krita-artists.org**

Enable Pen Settings to see how it works with a normal mouse. We find the default straight pressure setting kicks the brush size in too quickly. Click the simple up curve – the seventh icon along above the graph. You should find this eases in the brush size increase more subtly with applied pressure. If you find that this still isn't graduated enough, add a second control point to the curve and adjust the curve accordingly.

To see how different inputs can have interesting effects here, select the Speed option and deselect Pressure. When you start to use multiple inputs, you might need to deselect the Share Curve option under the graph, because these different physical inputs have wildly different attributes and ways that we want them applied.

On the Speed graph, select the first straight line preset. If you try drawing with this, you'll find the brush line stays very small, no matter how fast you move your pen. Add another control point to the line and drag it to about 10% on the X axis – this should result in a near vertical line up. If you now try doing some fast brush strokes, you'll get a natural-looking teardrop shape, like you would from an oil-paint brush stroke.

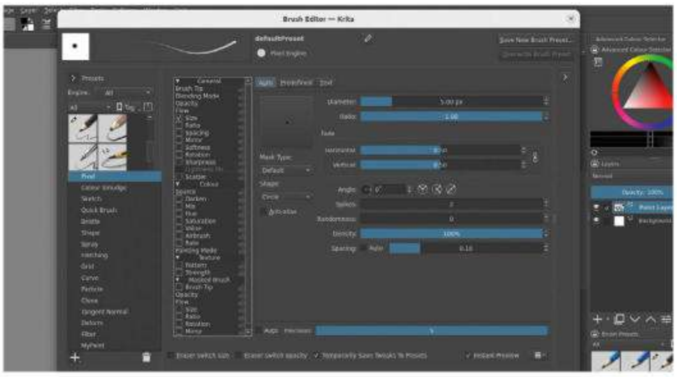
What's happening here is that the Speed option means that as you accelerate the pen from stationary, the brush quickly goes from being very thin to thick, then at the end of the stroke, as the pen is lifted, it reduces rapidly in size, giving the teardrop shape.

Try enabling the Pressure option again and give the brush a test. The two inputs start to have a really interesting end effect, with the dual inputs of speed and pressure applied to the brush size. On top of this, the Opacity option is also in effect, which controls the opacity of the stroke with pressure. You can select Opacity and deselect Enable Pen Settings to see what the plain brush strokes look like. Hopefully, this gives you an idea of how *Krita* offers massive amounts of creativity and customisation when it comes to brushes.

Playing around

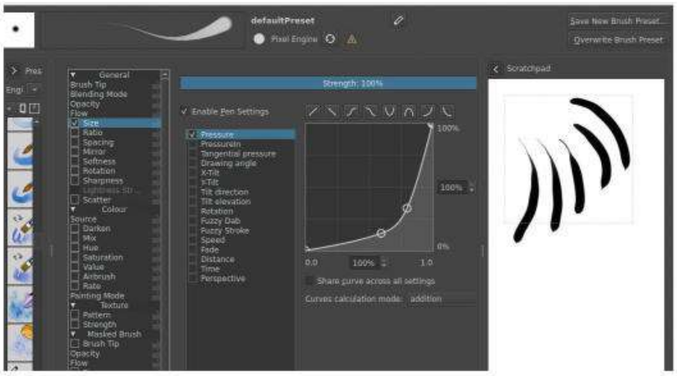
We can quickly have some fun by adding in a texture and utilising some randomising options. Select the

CREATING A KRITA BRUSH



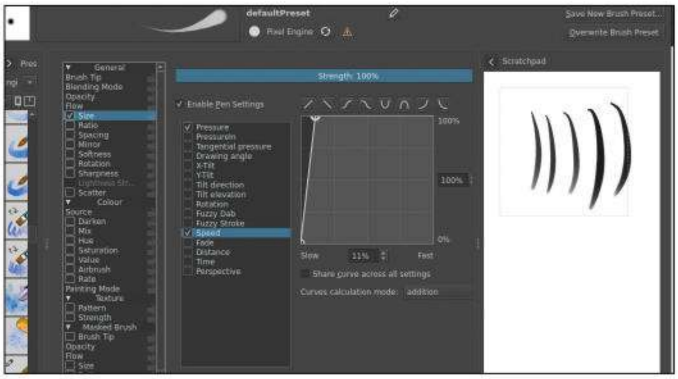
1 Create a new brush

Press F5 at any point to open the Brush Editor dialog. Open the left-hand Brush Presets and click the bottom-left + button and select Pixel. This creates a default Pixel Engine brush. Select Brush Tip and increase the Diameter to around 80, so we can easily see our adjustments.



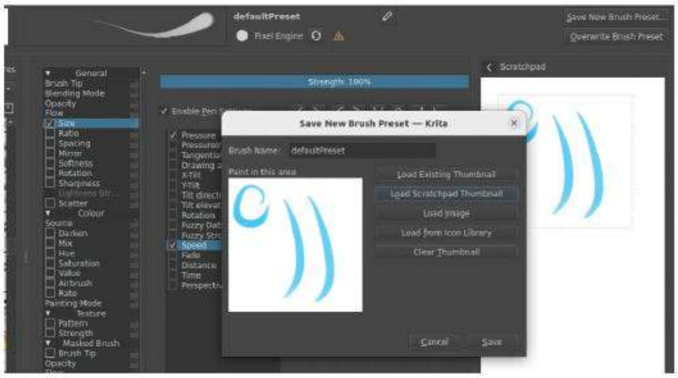
2 Under pressure

To create a pressure-sensitive brush, click the Size option – this should already be active – and select Pressure. This opens the standard user-definable graph. Deselect Share Curve Across All Settings or we'll lose any changes later. Add control points on the X axis at 60% and 80% to create a smooth up curve.



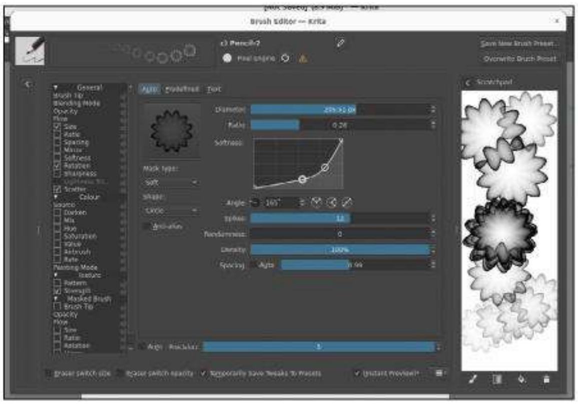
3 Speed editing

Open the Scratchpad on the right and give your brush a test run. Under Size again, select the Speed input and change the Curves Calculation Mode to Addition, so the various inputs are accumulative. We want the effect to kick in quickly, so add a control at 10% on the X axis – this should go straight to 100%.



4 Preview and save

Before we save the brush, clear the Scratchpad and do a neat stroke in the marked area. Click Save New Brush Preset. It takes the default icon, but give the brush a name first. Load Scratchpad Thumbnail uses the stroke you just did, else click Load From Icon Library, select one, click OK and add a stroke.



A world of interesting effects can be unlocked in the Brush Editor.

Pattern option and you'll see it's just that, adding a preset texture pattern to the brush stroke. This is working alongside the Speed and Size options we've already defined. By default, the pattern multiplies when drawn, largely drawing solid black. We can ease back on this by selecting the Options tab and choosing Subtract under Texturing Mode.

This should already be drawing a pretty pleasant texture, but it's very symmetrical, which might not be desirable, as it can overwhelm in large areas. Try adding the Rotation option – we found adding a curve with a control point around 35% worked well. This rotates the texture depending on the pressure applied. You could try swapping this for time or distance settings to mix things up. The other option we tried was Scatter. As it suggests, it scatters the brush based on the input. We found we had to set a very flat line – the control point was down at 5% on the Y axis. Just adding in those additional options transforms our base brush, though you should still be able to see the underlying Size option effects we originally created.

More tips

We dived straight in and skipped over what you might think would be the starting point: Brush Tip, the first option in the General section. The default round brush tip is fine for playing around, but now we're running out of space, it's worth highlighting, so select Brush Tip.

Under the Auto tab, Mask Type Default and Gaussian provide Shape options of Square and Circle, with Gaussian giving softer edges. We'll look at the Soft Mask Type shortly. Define the size with Diameter and how soft or hard the edge is with the Fade settings; click the chain button to apply this to just one set of sides. If you want a non-square tip, such as an ellipse or rectangle, adjust the Ratio setting. Once you have, you can use the Spike setting to add undulations to a circle tip or turn the square into a polygon shape.

The Soft setting provides a user-configurable Softness graph to define the brush tip. It's best to have a play with a line – along the top defines an all-solid brush, along the bottom would be an empty brush. Curving up from the bottom-left would be largely an outline brush graduating in from the centre.

If you've created a brush you're happy with, you can save the Preset using the top-right Save New Brush Preset option (just below is the Overwrite option to update an existing one). Clicking that opens the Save dialog. Here you're able to name your brush and assign a custom icon so you can spot it easily. Load Existing Thumbnails presents a list of existing thumbnails, oddly enough, from the other brushes – quick but not that great for spotting your brush.

If you've drawn an example line or two in the Scratchpad, Load Scratchpad Thumbnail is useful as it presents a real example of what the brush does. Load Image is any old image, while Load From Icon Library is most useful as it offers the various generic *Krita* pen and brush icons. You're then able to add an example line in the preview to match the default style of the other *Krita* brushes. Click Save and you're done. LXF

» USING AN ANDROID TABLET

You might already have something you can use as a graphics tablet: an Android device. They don't even need stylus support, as there are lots of cheap capacitive pens that enable you to use touch-sensitive screens – ideal for cold days when you want to keep your gloves on.

If you just want something to play with, see the GfxTablet project at **https://github.com/rfc2822/GfxTablet**. This turns your device into an absolute network pointing tool with stylus and pressure support, if the screen supports that. It's not too involved to get up and running. You'll want the *GfxTablet* Android app from F-Droid – install that and under Menu > Settings > Recipient Host, enter your PC's IP. Don't know the IP? In a terminal, type `ip address`.

Download the executable from GitHub – see the Readme link under Part 1, starting 'Networktablet'. Open a terminal, `cd ~/Downloads` and give the file executable permissions with `chmod a+x ./networktablet`. You can run this with `sudo ./networktablet` and your Android device should spring to life at this point.

Two caveats: under Wayland, the cursor disappears under Ubuntu, but you can force a X session that removes this issue:

```
$ sudo nano /etc/gdm3/custom.conf
```

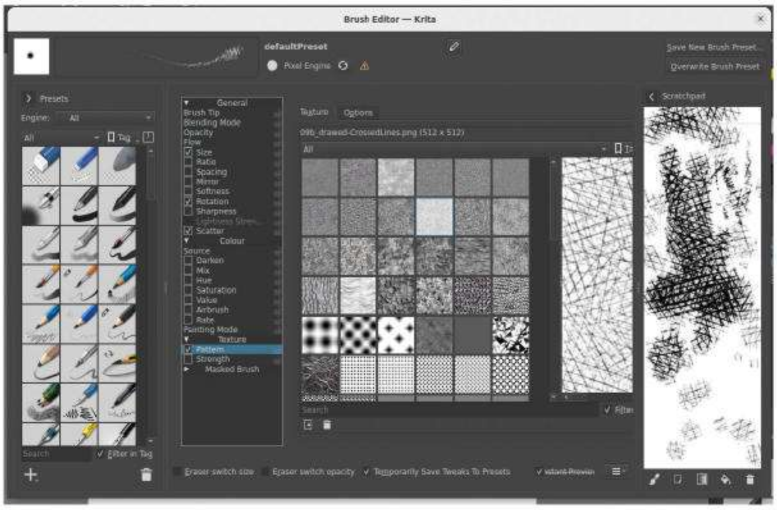
And remove the comment # symbol from this line:

```
# WaylandEnable=false
```

Press Ctrl+O to save (O for output) the file and Ctrl+X to exit. Reboot and you should be good to go.

QUICK TIP

You can still follow this tutorial with using just a mouse, everything will still work it's just the pressure sensing aspects will be lost, jumping to full pressure for a mouse press.



Add extra texture to your new brush with a redefined pattern.

» **MORE OPEN SOURCE CREATIVITY** Subscribe now at **http://bit.ly/LinuxFormat**

Protect your data with strong encryption

When **Stuart Burns** was told to lurk around the data centre, he brought a packed lunch and his laptop running Ubuntu 24.04.

This guide assumes your primary installation disk is already encrypted. If this is not the case, you will have to re-install the OS and when the installer gets to the Disk Setup page, select Advanced Features and choose the Encrypt With LVM option. It asks you for a passphrase and offers to set a backup key on the disk being encrypted, to be moved somewhere else later (if you forget the passphrase). Picking a good passphrase is critical. If you want to get nerdy, the EFF has a take on creating a secure passphrase with dice: **https://bit.ly/3y2wHGI**

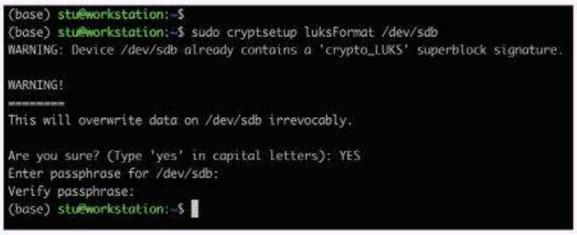
It can't be stressed strongly enough how critical it is to have a backup of this and a good passphrase. Losing it means reinstallation and no chance of data recovery.

Assuming the OS disk is encrypted, the encryption of additional disks is quite straightforward. The first step is to ensure that the **cryptsetup** package is installed on your system.

This package provides the necessary tools for managing encrypted volumes and LUKS encryption. On Ubuntu use `sudo apt-get install cryptsetup -y` or other *Apt*-based package systems.

Once the partition has been prepared, initialise (create) LUKS encryption on the partition using the *cryptsetup* command. Replace **/dev/sdX** with the appropriate partition identifier. Quadruple-check the right disk is chosen. To check, it's advisable to run `sudo lsblk` to show all the block devices connected to the system and their properties. Once the correct disk has been verified, use the following command to prepare the LUKS encrypted disk:

```
$ sudo cryptsetup luksFormat /dev/sdX
```



```
(base) stu@workstation:~$
(base) stu@workstation:~$ sudo cryptsetup luksFormat /dev/sdb
WARNING: Device /dev/sdb already contains a 'crypto_LUKS' superblock signature.

WARNING!
========
This will overwrite data on /dev/sdb irrevocably.

Are you sure? (Type 'yes' in capital letters): YES
Enter passphrase for /dev/sdb:
Verify passphrase:
(base) stu@workstation:~$ ▌
```

Using the LUKS format parameter to create a LUKS encrypted partition. Do note the note about the capital YES.

You are prompted to confirm the encryption (you need to type `YES` here), set a passphrase and verify it. Choose a strong passphrase and keep it secure, as it will be required to unlock the encrypted volume. Understand that this is a destructive operation and erases any contents from the disk or partition selected!

It's important to understand that every encryption command we use *(*`luksOpen`*,* `luksClose`*,* `luksformat`*)* is a parameter (or simply switch) of the *cryptsetup* command. Once you get that it's essentially `crypsetup lukOpen`, `cryptsetup lukClose` it is much easier to understand conceptually.

After initialising LUKS encryption, open the encrypted container using `cryptsetup luksOpen`. This can be thought of as unlocking the disk.

```
$ sudo cryptsetup luksOpen /dev/sdX encrypted_
volume
```

» EXTERMINATE! EXTERMINATE!

If you have an SSD with data on it that needs to be erased, there is a built-in function do to just that. This is known as secure erase mode. It does several things, including erasing the key that is stored on the disk (so it can never be recovered, as the key is missing). It also makes sure to erase the caching on the disk and any metadata as well. Each vendor implements this a bit differently but there is a tool to perform the erase in Linux, called *nvme-cli,* installed with `apt-get install nvme-cli`.

Using this command, show available disks via `sudo nvme list` to show all NVMe devices. To actually securely erase it, the command is `sudo nvme format -s1 /dev/nvme0n1` – subsitute **nvme0n1** for whatever drive is to be erased. Again, use caution and make sure the right drive is selected. It is critical that this is done correctly or it could render your machine unusable by erasing the EFI data.

If you are securing your disks because they were not encrypted before, it is a good idea to do a secure erase. What you can do is boot the install media from Ubuntu, run `try Ubuntu` and use a session to install and run the *nvme-cli* command. It saves the catch-22 of trying to format a mounted disk.

Replace **/dev/sdX** with the encrypted partition identifier and **encrypted_volume** with a name of your choice for the unlocked volume. Once the encrypted volume is unlocked, create a filesystem on the decrypted device using *mkfs.ext4* to format the partition to ext4:

```
$ sudo mkfs.ext4 /dev/mapper/encrypted_volume
```

This command creates an ext4 filesystem on the decrypted volume. Mount the encrypted volume to a directory of your choice using *mount*:

```
$ sudo mkdir /media/encrypted
$ sudo mount /dev/mapper/encrypted_volume /media/encrypted
```

Replace **/media/encrypted_data** with the desired mount point. One item often overlooked is that by default **/media/encrypted_data** is owned by root. A quick fix for this would be to set the ownership and the permissions to allow the normal non-root user (stu here, replace that with your own user) to access them. Do this by using the following command:

```
$ sudo chown stu:stu -R /media/encrypted
$ sudo chmod 775 -R /media/encrypted
```

You can now access the encrypted volume like any other disk partition. Store sensitive data within the mounted directory, and it is automatically encrypted on disk. To unmount the encrypted volume, use *umount:*

```
$ umount /mnt/encrypted_data
```

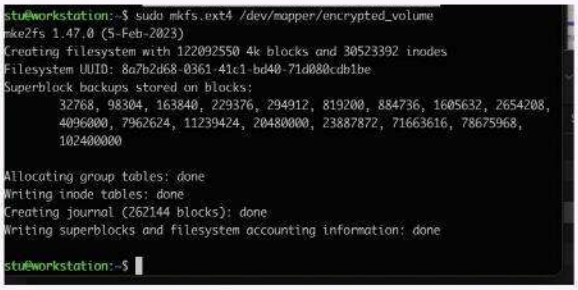
Close the container and lock the encrypted volume:

```
$ sudo cryptsetup luksClose encrypted_volume
```

Disk encryption, when the basic concepts are understood, is simple. The best time to encrypt all your non-encrypted items is yesterday. Failing that, today.

Whilst typing in the decryption key and mounting the disk is a bit long winded, there is a way to auto-mount the encrypted partitions. This is called *crypttab* and it allows for the auto-decryption of the partition or disk in question and also to mount it as well.

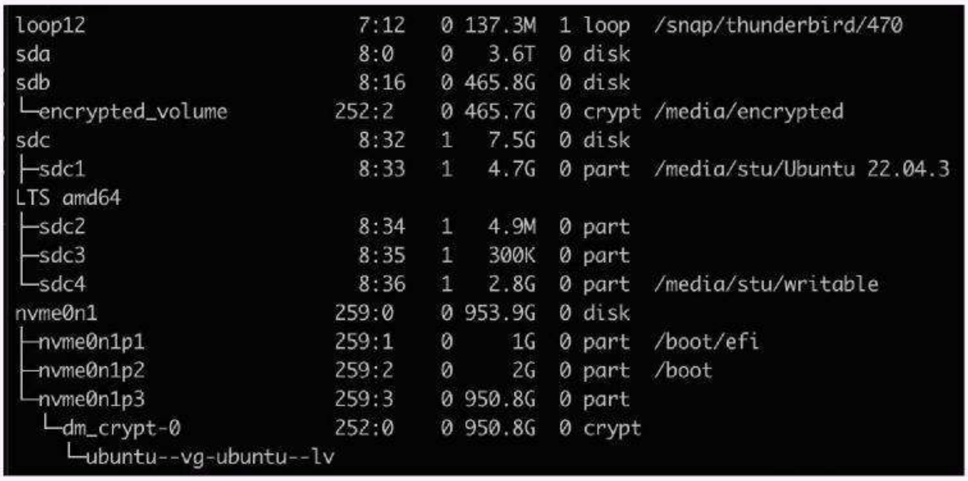
It works as a chain of trust, with a digital key stored on your encrypted boot disk, and uses that key (file) to unlock the additional encrypted partition. It is secure because to mount that disk, the attacker would need to be able to get to the key on the already encrypted disk. A detailed walkthrough of setting this up can be found at **https://bit.ly/3wquGUi.** LXF



```
stu@workstation:~$ sudo mkfs.ext4 /dev/mapper/encrypted_volume
mke2fs 1.47.0 (5-Feb-2023)
Creating filesystem with 122092550 4k blocks and 30523392 inodes
Filesystem UUID: 8a7b2d68-0361-41c1-bd40-71d080cdb1be
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000, 7962624, 11239424, 20480000, 23887872, 71663616, 78675968,
        102400000

Allocating group tables: done
Writing inode tables: done
Creating journal (262144 blocks): done
Writing superblocks and filesystem accounting information: done

stu@workstation:~$ 
```

Formatting the LUKS encryption container to a format that is understood by the Linux desktop.

Use the lsblk command to show all block devices seen by the system.



```
loop12               7:12   0 137.3M  1 loop  /snap/thunderbird/470
sda                  8:0    0   3.6T  0 disk
sdb                  8:16   0 465.8G  0 disk
└encrypted_volume  252:2    0 465.7G  0 crypt /media/encrypted
sdc                  8:32   1   7.5G  0 disk
├sdc1                8:33   1   4.7G  0 part  /media/stu/Ubuntu 22.04.3
LTS amd64
├sdc2                8:34   1   4.9M  0 part
├sdc3                8:35   1   300K  0 part
└sdc4                8:36   1   2.8G  0 part  /media/stu/writable
nvme0n1            259:0    0 953.9G  0 disk
├nvme0n1p1         259:1    0     1G  0 part  /boot/efi
├nvme0n1p2         259:2    0     2G  0 part  /boot
└nvme0n1p3         259:3    0 950.8G  0 part
  └dm_crypt-0      252:0    0 950.8G  0 crypt
    └ubuntu--vg-ubuntu--lv
```

Stuart Burns is a Linux administrator for a Fortune 500 company specialising in Linux.

» LUKS BEFORE YOU LEAP!

Recently, one of my mechanical hard disks started to fail after only four months. Sending it back was easy, and I knew that the data on it was safe because it was fully encrypted with LUKS encryption.

If anyone tried to get a sneak peek of the disk, they wouldn't be able to see much other than seemingly random data.

Of course, it is critical to use good encryption before you actually need it, because in these modern times of NVMe disks, there is no fully proven way to overwrite all the blocks, given the way that error correction works on solid-state disks. Similarly, a failing disk can not be guaranteed to be erased effectively because write access can be locked out, with read access left active.

LUKY CHAP

Created in 2004 and standing for Linux Unified Key Setup, LUKS is a disk encryption specification that provides a platform-independent standard for securing data at the disk level. It enables users to encrypt entire partitions, ensuring that data remains encrypted when it is stored on disk. LUKS operates by creating an encrypted container, known as a LUKS volume, which can then be mounted and accessed like any other disk partition.

It is very secure and includes strong, mathematically proven, open source encryption using the well-understood AES encryption protocol.

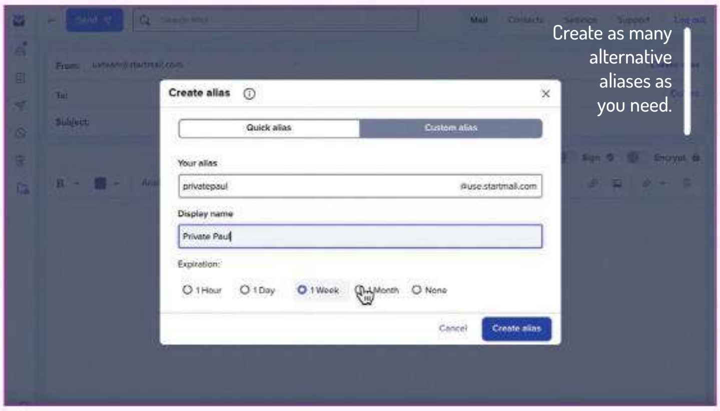
You can find out more about how to encrypt your disks using LUKS in the guide on the left.

StartMail

A man who trusts no one, **Krishi Chowdhary** demands ad-blocking and third-party trackers from his secure email. Looks like he's found them.

IN BRIEF

StartMail stands tall as a revolutionary email management platform in what most will say is a tough industry, given the sheer amount of market share that Gmail and Outlook hold.



Create alias

Your alias: privatepaul @use.startmail.com
Display name: Private Paul
Expiration: ○ 1 Hour ○ 1 Day ● 1 Week ○ 1 Month ○ None

Create as many alternative aliases as you need.

StartMail is a popular email service provider with a focus on security, privacy and email management. Founded in 2013, the platform can be used independently or in collaboration with other services such as Apple Mail or Outlook. Its primary purpose is to amp up encryption, leak-proof your email contents, and give you more control over how your emails travel.

In addition to the individual benefits of each plan, some features are common to both, such as 20GB of storage, email management, a built-in ad tracker and blocker, one-click email deletion, and end-to-end encryption for all emails.

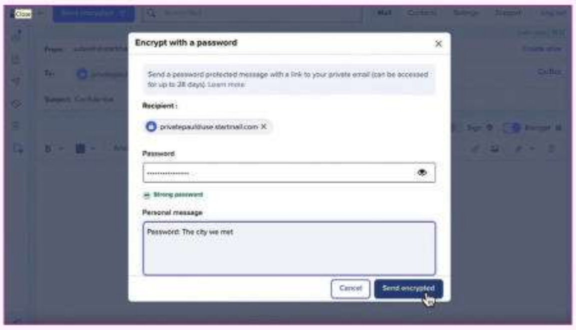
The Personal plan costs $5 per month when billed annually and lets you pay in Bitcoin (when you opt for annual billing). It also offers a seven-day free trial. You can create unlimited burner email IDs for temporary use and delete them all at once with a single click if/when needed. The only thing you might not like about this plan is that all your emails contain the StartMail domain – **yourbusinessname@startmail.com**, for example. Priced at $5.85 per month (annually), the Custom Domain plan lets you create your own domain.

One of the best things about StartMail is that it lets you create multiple aliases, so when you need to log in to a new site, you can use one of your aliases, get the job done, and dispose of that email.

It also offers group subscriptions, where you can add family members or friends to your account. The best part about this is that everyone gets their own separate inboxes and none of the emails overlap. Plus, instead of paying for each plan separately, you only have to pay a discounted rate for each additional inbox.

StartMail offers password protection and is pretty simple to use. Just toggle the switch on your email screen that says Encrypt, then click on the blue Send Encrypted button. A new window pops up where you're asked to set a password as well as a password hint.

Along with password-protected emails, StartMail has a bunch of other email security features that together provide world-class, unbreakable encryption. For instance, you can add a PGP signature at the end of your email. Using PGP, you can also encrypt your email in just one click and ensure that only the intended recipient is able to read the contents.



Encrypt with a password

Send a password protected message with a link to your private email (can be accessed for up to 28 days). Learn more

Recipient: privatepaul@use.startmail.com

Password: ••••••••••••••••

Strong password

Personal message: Password: The city we met

Encrypt your emails with basic passwords or power up with classic PGP.

If you're moving from a different email service provider, migrating and merging contacts only takes a few clicks. And the best part is that it works with all your favourite email apps, such as *Thunderbird, Outlook* or *Apple Mail*.

Another reason to praise StartMail's security is how it automatically blocks tracking pixels, which can be used to track your email activity. It also has built-in malicious link protection, which shows you the full web address before you click on an external link. Plus, it hides your IP address in email headers so that you can't be tracked by any third party.

What sets StartMail apart is its delightful combination of email management and security. One of our favourite features is Email Aliases. In a world where every website encourages you to share your email in order to access its content, StartMail ensures that you can keep your actual contact details private by enabling you to create burner email IDs. LXF

VERDICT

DEVELOPER: StartMail.
WEB: www.startmail.com
PRICE: $5 per month Personal ($59.95 per year)

FEATURES	8/10	EASE OF USE	9/10
PERFORMANCE	9/10	VALUE	7/10

The perfect email service provider if you prioritise security and privacy. You can set up burner emails, encrypt messages, create domains, and enjoy 20GB of storage.

» Rating 8/10

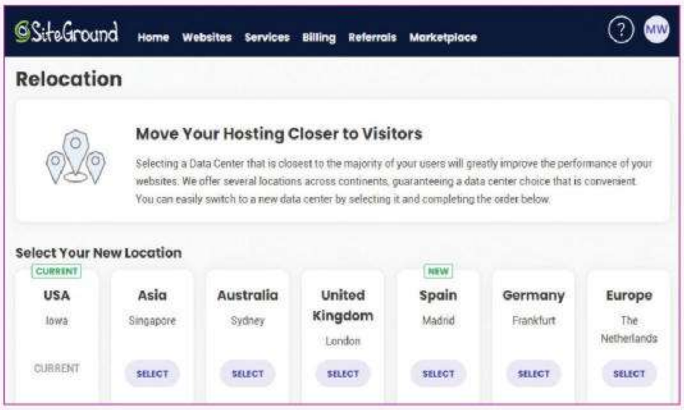
CREDIT: StartMail

SiteGround

Always demanding more and more features, **James Capell** suddenly wonders whether he's willing to shell out for them!

IN BRIEF

SiteGround is great for speed and reliability, and is good value for money if these are your priorities. Storage comes up a little short compared with other hosts for the same money. A strong contender for those looking for fast loading times all around the globe.



Relocation

Move Your Hosting Closer to Visitors

Selecting a Data Center that is closest to the majority of your users will greatly improve the performance of your websites. We offer several locations across continents, guaranteeing a data center choice that is convenient. You can easily switch to a new data center by selecting it and completing the order below.

Select Your New Location

Locate your server in any territory SiteGround offers; you must pay to move it later.

Now 20 years old, SiteGround is one of the world's largest independently owned hosting companies. It started off with a handful of staff and has grown to have 500 employees and data centres across six continents used by over 3,000,000 domains. Its products are split between web hosting and cloud hosting, with reseller and email marketing plans.

We're looking at the web hosting, which comes in three tiers: StartUp, GrowBig and GoGeek. The lowest plan, StartUp, supports one website, 10GB of web space, 10,000 monthly visits, a free domain, free SSL, daily backups, free CDN, free email and more. For new signups this costs £2.99 per month for 12 months, then renews at £13.99 per month. GrowBig (£4.99 per month) is essentially the same but supports unlimited websites and 100,000 monthly visits with faster PHP, while GoGeek bumps the monthly visits to 400,000 and costs initially £6.99 per month.

There's a Migrator plugin that provides an easy way to import your *WordPress* site from another host. The AutoUpdate system keeps your *WordPress* installation and plugins up to date, and experts will appreciate *WP-CLI*, a command-line tool that allows automation of many common *WordPress* management tasks.

Uptime.com accessed our site every five minutes over 14 days, recording any failures and how quickly the server responds. SiteGround had no fails, giving it a perfect 100% uptime. Average response time was 0.207 seconds, the fastest in our last 15 tests.

GTmetrix measures load speed by grabbing a test page on our site, and measuring how long it takes to display the main content, known as Largest Contentful Paint, or LCP. SiteGround's LCP was fractionally below average at 0.735 seconds, ninth fastest in our last 15 tests. But that's not far behind the top providers, and an acceptable time overall.

K6 goes beyond an individual load time by unleashing 20 simultaneous visitors and watching to see what happens. Our site managed an average of 14 requests per second, with a peak of 20. Again, that's fractionally behind the competition, but not enough that you're likely to notice.

Getting started with a web host can be intimidating, but SiteGround does a fair job of helping out. Log in to its Control Panel for the first time and you find links to support pages covering several common setup tasks: how to import an existing *WordPress* site, launch *WordPress*, create email accounts at a new domain, point an existing domain to *WordPress*, and more.

Launch *WordPress* and a wizard prompts you to choose a theme, then offers to install useful free plugins (contact forms, an image gallery, a calendar, Google Maps, WooCommerce, a contact manager, SEO advisor and more).

You can contact the support team via phone, live chat and ticket, although the website makes this more difficult than usual. There's a Contact Us button, but this walks you through a wizard that works hard to direct you to a support site article or website tool. We chose live chat, an agent appeared within seconds, and gave us a clear answer immediately. We tried phone support with similar success. Impressive, but we'd like it if the website didn't try so hard to ration our access.

SiteGround has data centres in the UK, USA, Australia, Germany, Netherlands, Singapore and Spain. Sign up for a SiteGround plan and you can choose which country hosts your site. If your audience is mostly in one country, that's good news; choose a data centre near your visitors and they see better speeds.

The company offers very generous compensation if the network is down. If you only have 99.9% to 99% uptime, for instance – so maybe just 44 minutes over a month – SiteGround promises an additional 10% of monthly hosting credits. And you get another month of free hosting for every 1% of uptime lost. LXF

CREDIT: SiteGround

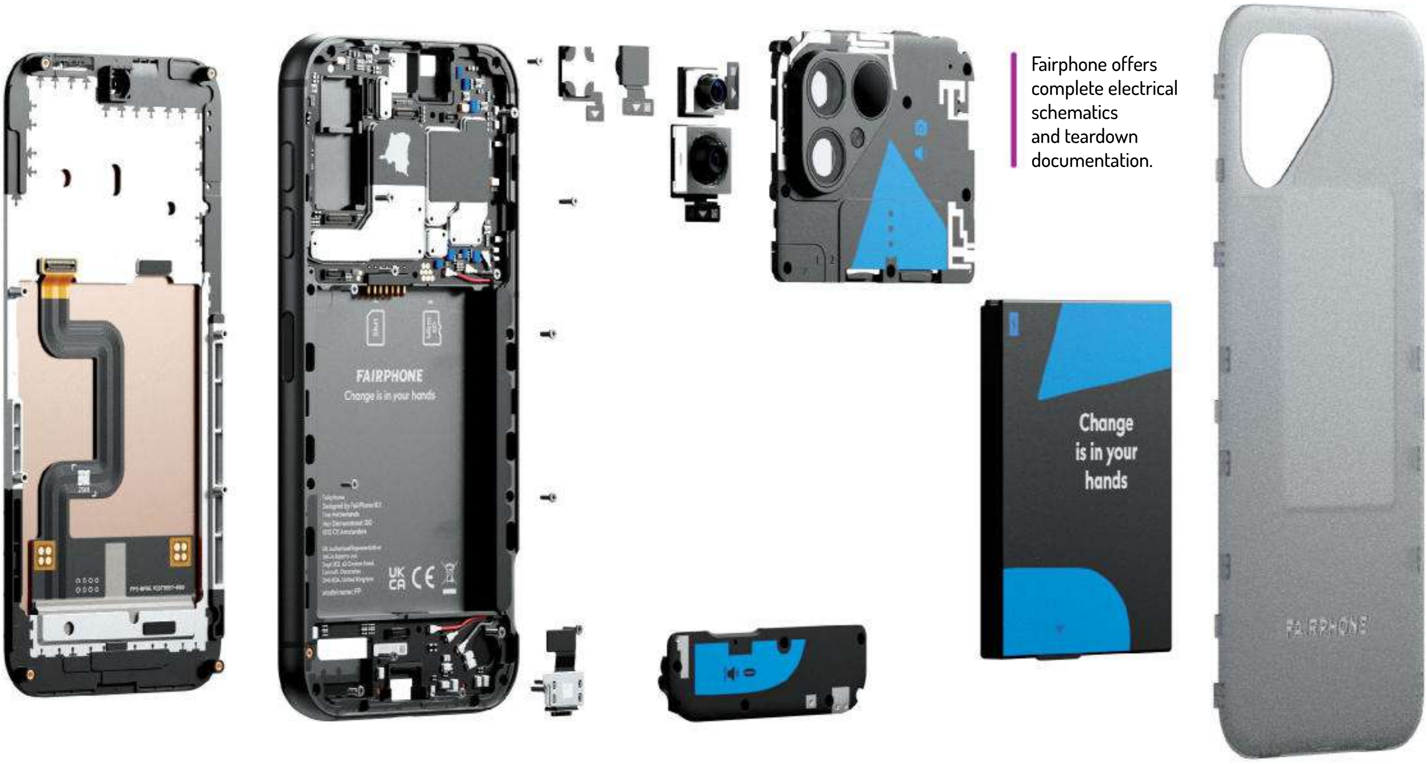
VERDICT

DEVELOPER: SiteGround
WEB: www.siteground.co.uk
PRICE: From £2.99 per month (then £13.99)

FEATURES	9/10	EASE OF USE	9/10
PERFORMANCE	8/10	VALUE	7/10

Great for speed and reliability, while storage comes up a little short, but if you can get by with fewer gigabytes, SiteGround should be seriously considered.

» **Rating 8/10**



Fairphone offers complete electrical schematics and teardown documentation.

Our green and pleasant phones

The environment is being destroyed by an addiction to devices. **Neil Mohr** investigates one company trying to turn the tide.

The figures are stark: 1.2 billion phones are made every year and there's just shy of 18 billion mobile devices in operation today, an incredible 2.5 devices for every human on the planet. What makes these figures worrying is the majority are manufactured in ways that make them effectively disposable consumables – if they're damaged or reach the end of their usable life, for many the most cost-effective solution is simply to bin them.

Certainly, recycling and reselling help extend device usefulness, but no matter how recyclable or repairable devices become, there's still a price that's paid beyond the device itself. Inside all digital devices are rare-earth elements, some of which – such as tantalum, tin, tungsten and gold, also called 3TGs – are known as conflict minerals, largely as their mining and sale are used to fund killings, violence and other human rights abuses in and around the areas they're mined.

It's such an issue that back in 2009, the US tried to introduce the Congo Conflict Minerals Act. While that failed, it started the ball rolling on forcing industries to publicly report their source of conflict minerals, with US legislation being introduced in 2010 – though part of it was struck down in 2014 by US courts for violating corporate free speech, if you can get your head around that. Intel was the first electronics company to publish details on conflict-free supply chains in 2012.

It's good to note that these legislations do have an effect, as Intel has continued to invest and evolve its supply chains to try to improve fairness, sustainability and corporate responsibility. Intel has a 2030 RISE (Responsibility Inclusivity Sustainable Enabling) goal and the aim of net-zero for its entire operation by 2040. You can read more at **https://intel.ly/49LhLdq**.

As crucial as it is for large industry players such as Intel to be involved in evolving something as complex and as messy as the raw-materials supply chain, it's different when it comes to delivering what you might consider to be an ethically sourced, manufactured and supported end device. That's where Fairphone comes in. The Dutch company formed in 2013 and debuted the Fairphone 1 smartphone in December that year.

The sheer scope of creating an ethical device is overwhelming when you consider all it encompasses. It's easy to focus on consumer-impacting features such as removable battery, replaceable modules and ultra-long warranties, but what about delivery, carbon-neutral energy generation, paying support staff, ewaste disposal, third-party working conditions and the environmental impacts of mineral extraction? If you're interested in just how deep this rabbit hole goes, Fairphone has a 94-page report for you at **www.fairphone.com/en/impact/** that outlines its challenges, accomplishments and opportunities to improve.

In a complex world, it's not possible to have direct control over everything you'd like, but Fairphone does have control over the repairability of its devices. When it comes to that, Fairphone is faultless, with the original Fairphone being the first and only phone to score a perfect iFixit 10/10 for repairability, an acclaim all its models can attest to, including the latest Fairphone 5.

Ear! Ear!

It's not been totally smooth sailing, though. Fairphone's first earbud attempt with the Fairphone True Wireless Earbuds scored 1/10. The good news is it has learned from that experience and its latest Fairphone Fairbuds are back to a perfect 10/10 for repairability, with easily replaceable batteries *(see the box, right, for more).*

The other factor Fairphone can exert itself on directly is the market. It's proved two important things: the first is that it is possible to manufacture an ethical phone, and secondly, there is indeed a demand and a desire to pay a premium for an ethically manufactured phone. Fairphone sold 120,000 phones in 2022, which was up 32% over 2021 figures, and in mid-2023, it entered the US market, which can only accelerate its overall sales, with the addition of a range of ethically built, repairable headphones to round out its offering.

Being ethical means you have to care about every aspect of your device from cradle to grave. Fairphone says than in Europe, users replace their phone every 2.7 years and around 700 million phones sit unused in drawers. Fairphone made the Fairphone 4 the first device to be ewaste neutral: for every phone and related module sold, another end-of-life phone or amount of ewaste is reused or recycled by Fairphone.

The area that Fairphone has least control over is the source of raw materials that are used to manufacturer the components within a Fairphone. Every material has a global journey, from extraction and processing to its transportation and component manufacturing, with the livelihoods of local workers and economies entwined. By Fairphone's own account, 40% of a Fairphone's total so-called focus materials are sourced fairly, highlighting the complexities involved.

The Fairphone XL headphones offer an ethical over-ear experience for your listening pleasure.

Fairphone is focusing on 14 core materials: aluminium, cobalt, copper, gold, indium, lithium, magnesium, nickel, plastics, rare-earth elements, silver, tin, tungsten and zinc. Moving raw materials supply to fairer mines or recycled sources is a complex global challenge. Some materials can be recycled: plastics, magnesium, rare-earth elements and tin. Some, mostly tungsten and aluminium, can be fairly mined through the supply chain, while in challenging areas such as gold mining, Fairphone is using credits to invest in affected communities.

Finally, Fairphone is directly improving workers' lives by paying them more and taking direct action to improve their employment circumstances. At the most basic level, Fairphone audits its suppliers on conditions, moving to worker testimonies. This leads to two direct actions, one being worker representation for tier one and two suppliers, alongside a living wage bonus, with Fairphone being the first company to implement such a scheme. The end effect is a $1.99 increase in the price of the Fairphone 4 to ensure a living wage bonus for factory workers. In 2022, this totalled $305,000 paid to 1,926 workers.

Now entering its 11th year, it seems nothing is going to stop Fairphone. With its fifth edition widely available *(see overleaf)* and newly launched Fairbuds to transform how we can ethically enjoy our music on the move, it seems there's no excuse not to help the cause and transform the industry by joining the fairest of phones.

» FAIRBUDS SOUND ETHICAL

Almost every single pair of earbuds on the market are tossed away when something goes wrong. If the battery stops holding a charge, you throw them away. Fairphone wants to fix that with the Fairbuds, which the company describes as the most repairable earbuds on the market.

Fairphone says you can replace seven different parts on the Fairbuds if something happens. If need be, you can swap the earbud battery, silicone ring, earbud ear tips, charging case outer shell, charging case core, charging case battery and the actual earbud.

As a bonus, not only are these repairable, but the company offers a three-year warranty. That means Fairphone is confident they won't break for an extended period of time, so you might not even need to use that repairability.

If you misplace one of your earbuds, the company will let you order a single replacement without having to buy a whole new set. This could save you a lot of money if you're prone to losing stuff.

While all the repairability and longevity are nice, you also want headphones that sound good. Fairphone includes some solid specs that rival other high-end models. There are 11mm titanium-coated drivers, three microphones per earbud, active noise cancellation (ANC), Bluetooth multipoint, Bluetooth 5.3, an IP54 rating, five hours of battery life with ANC on, up to 26 hours of battery life with the case, and an app for equalising sound.

At £129 for a set, they're not cheap, but with everything being replaceable and openly available from Fairphone's website, these are earbuds that are guaranteed to last. You just have to not leave them on a train by mistake…

We'd feel far more ethical with these plugged into our ears than any other brand.

CREDIT: Fairphone

Fairphone 5

Bram Lodewijks casts a meticulous and moral eye over the latest ethical phone.

The Fairphone 5 is far more than the sum of its replaceable parts.

SPECS

SoC: Qualcomm QCM6490 EL, 64-bit
Clock: 8-core, 4x silver 1.9GHz, 3x gold 2.4GHz, 1x prime 2.7GHz
GPU: Adreno 643L 812MHz
Mem: 8GB
Screen: 6.5-inch 90Hz OLED, 1,224x 2,700, 880 nits, Gorilla Glass 5
Storage: 256GB, up to 2TB microSD
Cameras: Main 50MP Sony IMX 800 OIS + EIS, UW 50MP Sony IMX 858 EIS; Front 50MP Samsung JN1 EIS
Video: 4k 30fps, 1080p 120fps, 720p 240fps
Comms: 4/5G (nano SIM+ eSIM), Wi-Fi 6, Bluetooth 5.2 LE, NFC, USB-C 3 OTG
Sensors: Facial, fingerprint, magnet, cccel, gyro, compass, light, proximity
Location: GPS/ AGPS, GLONASS, Beidou, Galileo
Battery: 30W QC3 4,200mAh removable
OS: Android 13
Warranty: Five years, IP55, MIL-810H
Size: 76x9.6x 162mm, 212g

While every other smartphone manufacturer bombards us with numerous new models every year, Fairphone plays differently. It has been more than eight years since we looked at the Fairphone 2 (**LXF210**), and this ties in with the Dutch company's overall sustainable approach: it wants to get rid of the culture in which you replace your smartphone every year.

The new Fairphone 5 continues the company's sustainable strategy. You are guaranteed to receive five Android updates and eight years of security updates, and Fairphone hopes to extend this to 10 years of updates. And where the Fairphone 4 had eight separate parts that could be replaced by an end user, the improved modular design now includes 10 parts that you can swap out and substitute with a replacement..

That's all great, but for the Fairphone 5 to be a success, it must still meet people's expectations for speed and quality. Especially when it costs £649 for the one model with 8GB of RAM and 256GB of storage; naturally, you can expand the internal storage with a microSD card, up to 2TB.

A modular approach

The most important aspect of the Fairphone 5's design, of course, is that it's modular. Nothing beats the nostalgic feeling of the plastic back that you can take off via the simple process of slipping your fingernail under a small recess at the back. The back is firmly fixed, though, and even if you drop the phone – which happened during testing – everything remains stuck. After that fall, there was no scratch to be seen on the screen, which is protected by Gorilla Glass 5.

This modular approach leads to a chunky design compared to normal phones. Fairphone has scraped off a couple of millimetres compared to the 4, but you will notice its 9.6mm thickness. Still, it fits well in the hand and a plastic back means it doesn't slip on smooth surfaces such as a glass table. The right-hand side still holds the power button with a built-in fingerprint scanner, but this works significantly better than on the Fairphone 4.

You might think that a smartphone with a detachable back isn't water resistant, but the Fairphone 5's improved IP55 rating means that it's safe from heavy rain or a shower jet. Don't submerse it in water, though. Finally, the new version with transparent back *(as seen in image below)* is a nice addition for users who (rightly) like to brag about their durable smartphone's ethical credentials, or you can choose sky blue and matte black variants. All made, naturally, from 100% recycled plastic.

Its plastic back ensures the phone doesn't skid about when placed on a smooth surface.

Big display upgrade

Then we come to the phone's screen, which is perhaps the biggest upgrade over the Fairphone 4. That previous model had an IPS display with a 60Hz refresh rate and a notch, while the Fairphone 5 includes an OLED display with 90Hz refresh rate and a punch-hole for the selfie camera. In addition, the screen bezels are smaller, meaning the display is now 6.4-inch instead of 6.3-inch, without the phone itself becoming any larger. The edge bezels are thicker than those of rival smartphones, but the top and bottom edges are the same size, so you end up with a fine, balanced design.

The OLED panel certainly means a wider colour gamut than before, but the screen isn't without its flaws. For example, certain shades of red appear darker than usual. Nor is there any support for HDR10 or similar standards. What you do get, however, is excellent brightness, with a maximum

Chunkier than most modern phones, the Fairphone 5 still fits nicely in your hand.

of 880cd/m^2 ensuring easy readability; it's only in the brightest summer sun that you might sometimes have trouble seeing everything properly.

Camera boost

Fairphone has also tinkered with two of the phone's cameras. The main camera has a larger sensor, which helps in low-light conditions, while the selfie camera now has a more impressive 50MP sensor instead of 25MP. The new primary and selfie cameras both produce excellent results. In natural light, the main camera produces sharp images with true-to-life colours and good dynamic range.

We do wish Fairphone had upgraded the ultrawide, though. The centre of these photos is fine; however, the edges show noise and sometimes even discolouration. The other issue we have with this side of things is over how slow the camera app is to respond. Hopefully, Fairphone will make some tweaks to help matters here. Videos, especially, take a long time to be saved, and you can't take a new photo or video while the message 'Save…' sits on the screen.

More power, please

The Fairphone 5 is powered by Qualcomm's QCM6490 chip, which the company chose for stability – it has guaranteed long-term updates – rather than speed. Unfortunately, it's a poor choice for even undemanding games such as *Yu-Gi-Oh! Duel Links.* This game doesn't falter on far cheaper smartphones, but after half an hour, the phone became hot and we started to see frame drops. We saw similar problems even on light games such as *Pokémon Go* and *Hearthstone,* and at one point the Fairphone 5 even got hot when we were watching TikTok videos.

So, gamers shouldn't choose this phone. For everyone else, though, the choice is more nuanced. *Geekbench 6* scores of 1,131 and 2,985 (single-core and multicore respectively) are roughly equivalent to Android phones costing around £300, and compare to 1,401 and 3,368 for the Google Pixel 7a. The Fairphone 5 has enough power for today, overheating problems aside, but this phone is likely to feel slow by 2028.

5G is there, as well as support for Wi-Fi 6. While the Fairphone 5's performance isn't fast, its internet connection should be fine throughout its life. The other good news is that Fairphone won't weigh you down with bloatware – there's the *Fairphone* app and that's it. And now is a good point to emphasise that you're guaranteed to get updates from the supplied Android 13 to Android 18. With security updates promised until September 2031 at the very least, no other phone comes close for long-term support.

Fair conclusion

On the one hand, this phone is far too expensive for what you're getting. Performance is slow compared to similarly priced phones, there are overheating issues, and the battery is hardly a high-flyer.

On the other hand, it's a triumph. Almost every part, from the cameras and the screen to the battery and speaker, can easily be replaced with no more than a screwdriver. Furthermore, the cameras are solid and the screen has had a huge upgrade in the form of an OLED panel with a 90Hz refresh rate.

It comes with an unrivalled five-year warranty, along with software support well into the 2030s. And every Fairphone 5 you buy includes a £1.99 living wage bonus for the people who made it. Fairphone also invests in projects to reduce CO_2 emissions, has a transparent supply chain, and the assembly plant uses 100% renewable energy.

The final argument in the Fairphone 5's favour is that the price needs to be looked at over the course of years. If it lasts six-and-a-half years, for the sake of argument, that's £100 per year, which is likely to be far less than the phone it replaces. LXF

» BATTERY SWAPOUTS

Here's another big plus point: when the supplied 4,200mAh battery starts to deteriorate, it's simplicity itself to replace it. You may even choose to buy a second at time of purchase (this costs just £36), as this will not only extend each unit's life, but also give you the option to swap it out if the Fairphone runs dry.

The Fairphone 5 supports 30W fast charging, but there's no cable or adaptor in the box to avoid wastage. You can buy Fairphone's charger at time of purchase (£22), but because Fairphone uses the standard Quick Charge 3.0, almost every 30W charger will charge the battery optimally; it took about 30 minutes to go from empty to 50%. That won't win any prizes, but Fairphone focuses on the health of the battery rather than speed. Wireless charging isn't an option.

In truth, battery life isn't a strength. Perhaps that's why the screen is set to 60Hz refresh rates out of the box rather than 90Hz. With light to medium use, we eked out a full day's use, but if you scroll too much on TikTok or watch Netflix on your commute home from work, you will have to charge it in the evening. Games are the death knell for the battery – after barely an hour of *Hearthstone,* we lost almost 50% of charge.

VERDICT

DEVELOPER: Fairphone
WEB: http://shop.fairphone.com
PRICE: £649

FEATURES	9/10	EASE OF USE	9/10
PERFORMANCE	7/10	VALUE	8/10

The best Fairphone yet is also the most repairable, but don't expect the last word for speed or image quality.

» Rating 8/10

tom's HARDWARE

GET ALL THE ESSENTIAL BREAKING NEWS FOR THE TECH ENTHUSIAST!

No matter if you're building a PC, buying a laptop or learning about robots, Tom's Hardware has all the comprehensive knowledge you need.

Scan & Subscribe for free!

tom's HARDWARE
SIGN UP · SHARE
NEWS STREAM
RTX 4070 vs RX 7900 GRE faceoff: Which mainstream graphics card is better?
Ampere rides again in Nvidia's single-slot RTX A1000 and A400 for 'professionals'
Micro Center makes RTX 4090 owner a low-ball $700 offer on trade-in
'Enhanced' Nvidia A100 GPUs appear in China's second-hand market with more cores and more VRAM
AMD unveils Ryzen Pro 8000-series processors — Zen 4 and AI engines come to the commercial market
Asus ROG Swift PG49WCD 49-inch gaming monitor review: The most colorful OLED yet
Award-winning Chinese processor design company Hyasic Chip files for bankruptcy — firm's CPU, DSP, GPU, and AI processor IPs in limbo
MORE NEWS >

NEW
Everything you need to start coding with Python in Linux
The Python Book
Over 400 essential tips
50 ESSENTIAL PYTHON COMMANDS
Learn to use Python • Program games • Get creative with Pi

GET READY TO BECOME A TRUE PYTHON EXPERT

Make Python work for you with tutorials on coding with Django, Flask, Pygame and even more useful third-party frameworks.

ON SALE NOW

FUTURE

Ordering is easy. Go online at:

magazinesdirect.com

Or get it from selected supermarkets & newsagents

HotPicks

DigiKam » Haruna » Switcheroo » Wallpaper Downloader » Joplin » Remmina » NIS One-Click-Backup » Armagetron Advanced » Taisei Project » Varia » Midori

Mayank Sharma
hasn't compiled software in ages, and is surprised how quickly devs have taken to the three platform-independent package management formats.

IMAGE MANAGEMENT

DigiKam

Version: 8.3.0
Web: www.digikam.org

DigiKam bills itself as a professional photo-management app, but it does a nice job of dealing with them in such a manner that it appeals to all kinds of users. Although designed as part of the KDE desktop, the app works well on any desktop environment.

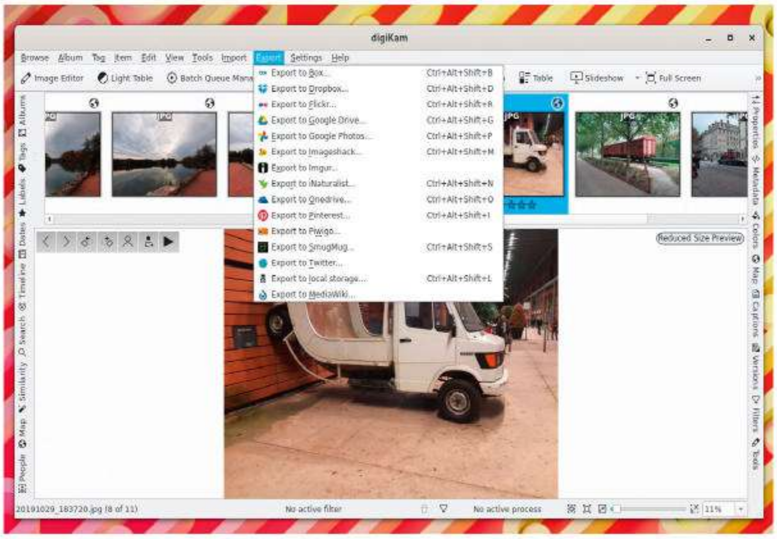
You can find *DigiKam* inside the official repos of virtually every distro, but the best option is to grab the official AppImage from its website. Once downloaded, give it executable permissions either using the file manager or with the `chmod +x` command. The app has a first-run wizard that asks you for the folder that houses your images.

Next up, the app asks you for the type of database and its location. By default, it uses the SQLite engine, and houses it in the same directory as the one you've selected in the previous step.

You're then asked to define how the app handles RAW images. By default, *DigiKam* directly imports these images into the app. Advanced users can ask *DigiKam* to pass these images to the RAW import tool, where they can tweak the images before importing them. There are several other steps that can help you define the app's behaviour. If you aren't a professional photographer, it's best to go with the default options.

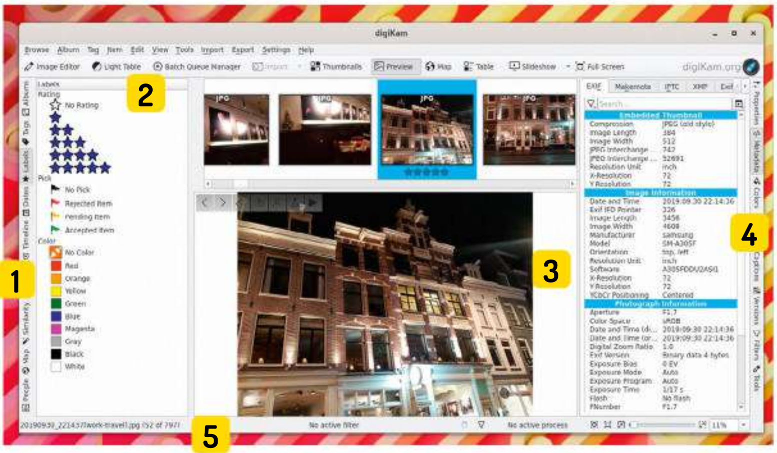
DigiKam also has a number of useful features such as face detection and more. However, these features aren't bolted on to the default installation. After you've run through the first-launch wizard, the app offers to download these for you. All told, these additional features weigh in at just under 900MB, so grabbing them can take a while, depending on your connection.

Besides sorting through vast collections, *DigiKam* also doubles up as a very capable photo editor. It offers a whole host of image-editing options that you can use to fix common issues such as red eye, colour and lighting with just a few clicks.



DigiKam has a whole lot of extensions to add to its already impressive feature list, and can also share images on popular image-sharing websites and social networks.

LET'S EXPLORE DIGIKAM...



1 Left toolbar
Use this toolbar to sort through images. Besides the list of photos, it can also sort your image collection by tags, labels, dates, faces and more.

2 Main toolbar
From here you can open additional tools such as the image editor, the light table to compare images, the batch queue manager to process multiple images, and more.

3 Image preview
A preview of the selected image is shown here. Right-click on the image to bring up a list of actions you can perform on it.

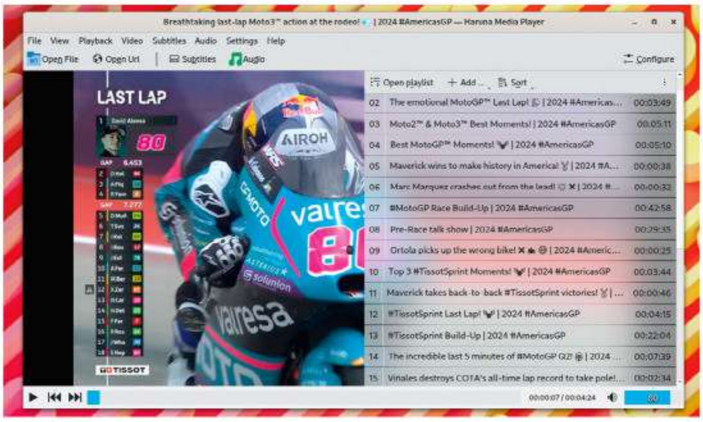
4 Right sidebar
This sidebar houses important info and actions for the selected images. Properties and Metadata display all kinds of details, and Colours displays histograms.

5 Status Bar
The status bar shows information about the open image, such as details about the number of filters, a trash icon to delete all filters, options to resize the preview, and more.

VIDEO PLAYER

Haruna Media Player

Version: 1.0.2
Web: https://haruna.kde.org


Breathtaking last-lap Moto3™ action at the rodeo! | 2024 #AmericasGP — Haruna Media Player

02 The emotional MotoGP™ Last Lap! | 2024 #Americas... 00:03:49
03 Moto2™ & Moto3™ Best Moments! | 2024 #AmericasGP 00:05:11
04 Best MotoGP™ Moments! | 2024 #AmericasGP 00:05:10
05 Maverick wins to make history in America! | 2024 #A... 00:00:38
06 Marc Marquez crashes out from the lead! | 2024 #... 00:00:32
07 #MotoGP Race Build-Up | 2024 #AmericasGP 00:42:58
08 Pre-Race talk show | 2024 #AmericasGP 00:29:35
09 Ortola picks up the wrong bike! | 2024 #Americ... 00:00:25
10 Top 3 #TissotSprint Moments! | 2024 #AmericasGP 00:03:44
11 Maverick takes back-to-back #TissotSprint victories! | ... 00:00:46
12 #TissotSprint Last Lap! | 2024 #AmericasGP 00:04:15
13 #TissotSprint Build-Up | 2024 #AmericasGP 00:22:04
14 The incredible last 5 minutes of #MotoGP Q2! | 2024 ... 00:07:39
15 Vinales destroys COTA's all-time lap record to take pole!... 00:02:34

00:00:07 / 00:04:24

VLC is the de facto media player on the Linux desktop. While it's very adept at playing all kinds of media, it's designed to do a lot more as well. So, if you want an app to just play audio and video files, there are lots of options, such as *Haruna*. Although it's built for the KDE desktop, the app is officially available as a Flatpak, which means you can run it atop any desktop environment.

Fire up a terminal and type `flatpak install flathub org.kde.haruna` to install the player. Once installed, the app is listed in your distro's application launcher.

Haruna is built as a front-end to the CLI-based *mpv* player, and one of its best features is its automatic playlist function. With this, the app automatically creates a playlist with other media files housed in the same folder as the one that's playing.

You can bring up the playlist by hovering over the right-side of the player window or by pressing the `p` key. The playlist overlays the playing video. The playlist has three self-explanatory display styles, namely Normal, which is the default, Thumbnail or Compact.

Most actions in *Haruna* can be triggered with keyboard shortcuts. Pressing Ctrl+` (backtick) brings up a searchable action palette that lists all shortcuts.

Another interesting feature is *Haruna's* ability to play YouTube videos, either from a specific link or from a YouTube playlist. If a YouTube video has chapters, these are displayed in *Haruna's* progress bar.

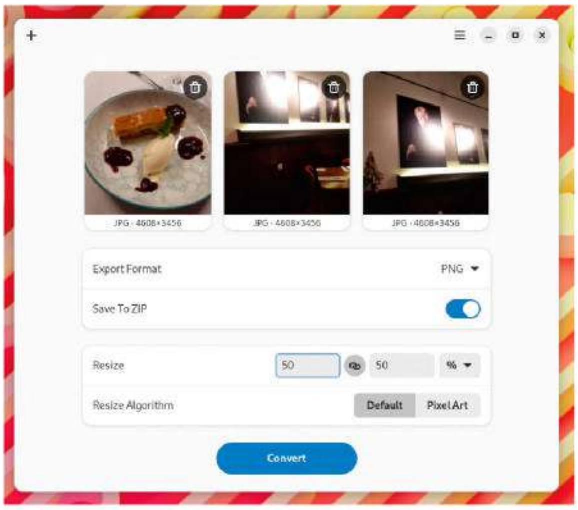
You can middle-click on the progress bar to jump to the next chapter of any video, online or offline. You can also tell the app to skip chapters containing specific keywords, which doesn't sound all that useful for the majority of us. However, it must have its uses, as it's one of *Haruna's* headline features.

As a KDE app, Haruna has a fleshed-out settings window from where you can tweak the app's interface, reconfigure the keyboard shortcuts and more.

IMAGE CONVERTER

Switcheroo

Version: 2.1.0 **Web:** https://gitlab.com/adhami3310/Switcheroo



Export Format PNG
Save To ZIP ●
Resize 50 50 %
Resize Algorithm [✓ Default] Pixel Art

One of the best image converters out there is the CLI-based *ImageMagick*. Even though that app is amazingly dexterous, CLI software isn't everyone's cup of tea. If you are one of them, you should take a look at *Switcheroo* instead, which offers some of the most-used functions of *ImageMagick* via a graphical front-end.

The app is officially distributed as a Flatpak, and you can install it with `flatpak install flathub io.gitlab.adhami3310.Converter`.

Switcheroo has a straightforward interface. You begin by giving it an image or even a bunch of them. You can either drag and drop the images, use the Open Images button to browse the filesystem, or ask the app to pick them up from the clipboard.

The app reveals further actions, depending on whether you have given it a single image or a number of them. If it has a single image, *Switcheroo* gives you a pull-down menu to export the image to another format. By default, the app offers common formats, such as JPG, PNG, WebP, and a handful of others.

You can also head to its hamburger menu and select the Show Less Popular Datatypes option to add a bunch of other formats to the menu including TIFF, BMP and more.

Next up, you get the option to resize the image. By default, the app resizes by pixels, but you can also ask it to resize by percentages. The app maintains the aspect ratio of the image by default, too, but you can untoggle the chain icon to specify your values.

Finally, you can change the resizing algorithm. The app doesn't specify which one it uses by default, but you can ask it to use the Pixel Art algo. If you aren't sure which to use, stick with the default option.

If you are working with multiple images, Switcheroo gives you the option to roll them into a ZIP file after it's done converting them.

AUTOMATIC WALLPAPERS

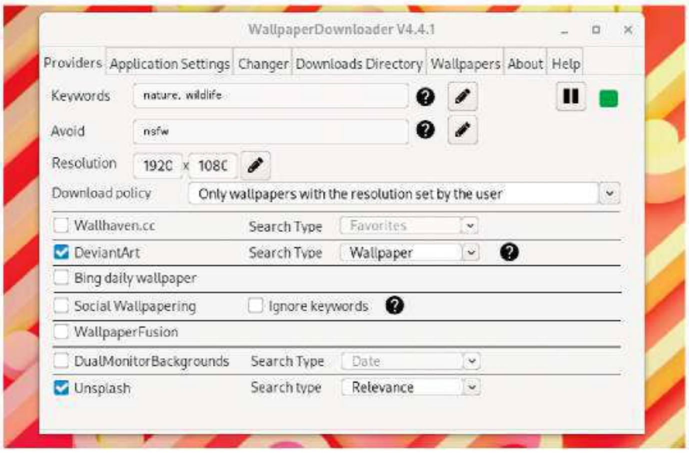
Wallpaper Downloader

Version: 4.4.1 **Web:** https://bitbucket.org/eloy_garcia_pca/wallpaperdownloader

Most distros roll in a bunch of wallpapers with their releases. And most of the time, you don't even see them as they are obscured by your open windows. But if you are fond of wallpapers and aren't satisfied with the default ones, *Wallpaper Downloader* can fetch and set wallpapers for you.

The app is available on Flathub, and you can grab it with `flatpak install flathub es.estoes.wallpaperDownloader`. *Wallpaper Downloader* has a multi-tabbed interface. In the Providers tab you toggle the websites from where you want to download the wallpapers. The app supports just over half a dozen sites, including DeviantArt, Unsplash and more.

By default, the app downloads any wallpaper, but you can use the Keywords field to limit it to those that match the keywords. If you don't specify any keywords, you can optionally use the Search Type pull-down to restrict wallpapers from one of the preset genres.



Keywords: nature, wildlife
Avoid: nsfw
Resolution: 1920 x 1080
Download policy: Only wallpapers with the resolution set by the user
☐ Wallhaven.cc — Search Type: Favorites
☑ DeviantArt — Search Type: Wallpaper
☐ Bing daily wallpaper
☐ Social Wallpapering — ☐ Ignore keywords
☐ WallpaperFusion
☐ DualMonitorBackgrounds — Search Type: Date
☑ Unsplash — Search type: Relevance

As its name suggests, Wallpaper Downloader houses the wallpapers in your computer and you can use the app to manually plaster them on your monitor any time.

The app downloads wallpapers in your monitor's native resolution, but you can ask it to download wallpapers in a custom resolution by specifying it in the Resolution field. Also review the Download policy and ensure it's set to download wallpapers that match the resolution you've set. By default, the wallpapers are stored in **~/.wallpaperdownloader/downloads**, but you can change that on the Downloads Directory tab.

Now move to the Application Settings tab, where you can change the frequency of how often the app downloads a new wallpaper, and the size of the download directory. Finally, head to the Changer tab, and specify a duration for the app to switch wallpapers. As soon as you set a duration, the app takes control and starts setting wallpapers based on your settings.

NOTE TAKING

Joplin

Version: 2.14.20
Web: https://joplinapp.org

What sets *Joplin* apart from the other note-taking and to-do apps, is that it offers both functions in one app through a neat-looking user interface.

The project recommends downloading the app through its installation script. In a terminal, type:

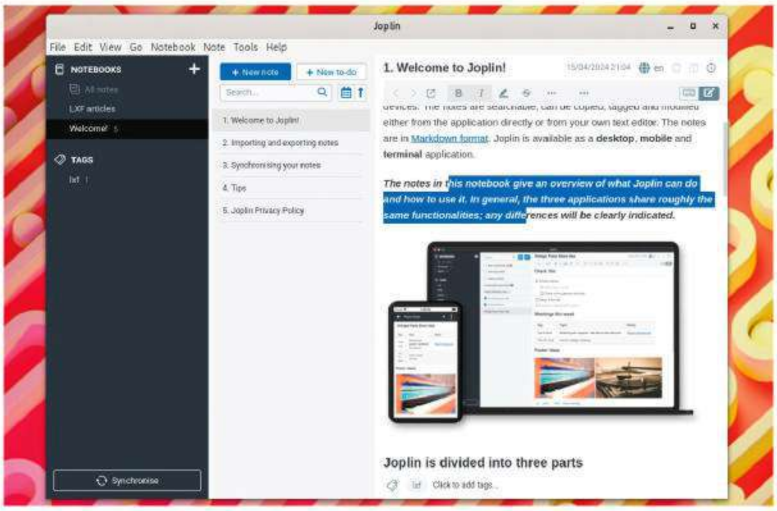
```
$ wget -O - https://raw.githubusercontent.com/laurent22/joplin/dev/Joplin_install_and_update.sh | bash
```

The script downloads *Joplin's* AppImage and sticks it to the Applications menu. The app opens a Welcome note on launch, that runs you through its main features.

Joplin's desktop interface is divided into three panels. The one on the left shows a list of notebooks and tags, its centre panel shows a list of the notes in the current notebook, and the right panel shows the selected note.

You can create as many notebooks as you like and nest notebooks inside each other. You can also give each note an emoji or custom icon so they stand out.

With *Joplin*, you have a choice of working with your notes in rich text or Markdown language, and you can switch between them at any time. The app has a two-pane interface for Markdown. You can type in the left-hand pane and see how it'll come out on the right.



1. Welcome to Joplin!
2. Importing and exporting notes
3. Synchronising your notes
4. Tips
5. Joplin Privacy Policy

devices. The notes are searchable, can be copied, tagged and modified either from the application directly or from your own text editor. The notes are in Markdown format. Joplin is available as a **desktop**, **mobile** and **terminal** application.

The notes in this notebook give an overview of what Joplin can do and how to use it. In general, the three applications share roughly the same functionalities; any differences will be clearly indicated.

Joplin is divided into three parts

Joplin supports dozens of plugins through which you can add all kinds of functionality to help manage your notes more efficiently.

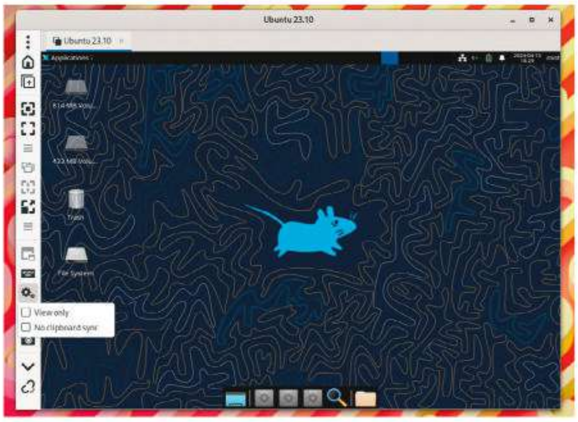
If you don't know what Markdown is and don't want to learn, you can use the rich text option, which gives you the standard WYSIWYG text editor. As well as text, you can also add any kind of file to the note, and even embed images directly into the note.

Joplin also lets you organise your notes with tags. Right-click on any note in the note list, and head to the Tags option. Then key in all the relevant tags for the note. These appear in the left panel underneath the list of Notebooks to help you sort through your notes.

REMOTE DESKTOP

Remmina

Version: 1.4.35
Web: www.remmina.org



Remmina can also make quick one-off connections; just select a protocol from the pull-down menu and key in the address of the remote machine.

A very usable remote desktop client, *Remmina* supports a wide range of protocols, and packs in a good number of features, which makes it a favourite with Linux users.

The app is officially distributed as a Flatpak and can be installed with the `flatpak install flathub org.remmina.Remmina` command.

Before you can establish a connection, *Remmina* asks you to create a profile to define its parameters. At the very least, you have to select a protocol from a drop-down list and enter the server's IP address.

If you'll be connecting to this machine often, it's best to hit the + button and create a profile. You have to give this connection a name, and optionally a label and a group to help distinguish it from other connections.

The other settings vary depending on the type of connection you're making. For a VNC connection, for instance, you get checkboxes to toggle some quick settings, such as starting a simple View Only session and more. Similarly, for an RDP connection, you can pre-bake the resolution for the remote desktop and choose a quality for the connection. For most connections, including both VNC and RDP, you can ask *Remmina* to execute commands before and after establishing the connection.

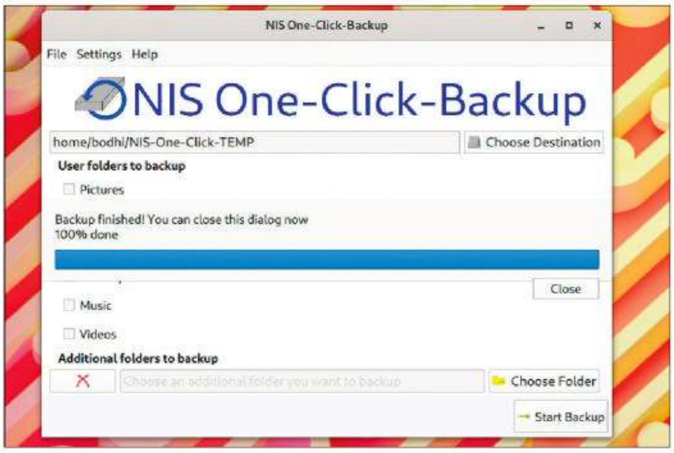
Remmina has a tabbed interface for managing multiple remote sessions from a single window. When connected, there are buttons for common tasks, such as switching to full-screen or scaled mode. Remote desktops with higher resolutions are scrollable/scalable in both window and full-screen mode.

The app also has a fleshed-out Preferences section. In here you can find options to tweak the app's behaviour and customise its user interface. You can also review and remap the default hotkeys for common tasks. Most importantly, this section houses several security settings to safeguard the app against misuse.

BACKUP

NIS One-Click-Backup

Version: 1.2.2.1 **Web:** https://gitlab.com/dev-nis/nis-one-click-backup-qt


NIS One-Click-Backup

home/bodhi/NIS-One-Click-TEMP

User folders to backup

☐ Pictures

Backup finished! You can close this dialog now
100% done

☐ Music

☐ Videos

Additional folders to backup

Choose an additional folder you want to backup

NIS One-Click-Backup is very basic and simple. It helps inculcate the habit of taking backups in users who don't because they perceive the exercise to be cumbersome.

Backup apps come in all shapes and sizes, and for all kinds of purposes. As its name suggests, *NIS One-Click-Backup* does exactly that; you use it to back up folders to external locations, in a single click (or thereabouts). It is available as a Flatpak and can be installed with `flatpak install flathub io.gitlab.dev_nis.one-click-backup`.

The app has a simple interface. By default, it lists the folders that most distros create under your **home** directory, such as **Documents** and **Downloads**. Toggle the checkboxes next to folders you want to back up.

Hit Choose Destination to point the app to the remote location where you want it to house backups. It doesn't support any cloud services or network protocols. You have to ensure the remote location is mounted and available through your file manager.

When you're done, smash the Start Backup button to back up the contents of the selected folders. The app creates a date-stamped folder under the destination, such as **Backup_13-04-2024**, and then simply copies the folders and their contents in there.

Besides the default folders, the app can also help you back up the contents of any other folder in your filesystem. Click Choose Folder and point the app to the folders you want to include in your backup.

The app is very simple and doesn't offer to compress the backups, or take incremental or differential backups. So, if you back up the same folder to the same destination on the same day, it simply replaces the contents of the previous backups with the new one. And true to its name, the app only takes backups, and there's no option that can help you restore the backed-up items.

ARCADE

Armagetron Advanced

Version: 0.2.9.2.3
Web: www.armagetronad.org

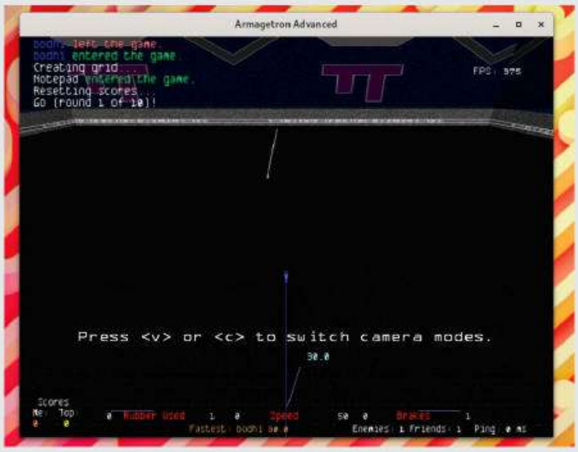
Armagetron was an arcade game based on the film *Tron,* centred around the movie's iconic light cycle. In the classic game, the player controls a light cycle, which leaves a wall behind it, turning only at 90-degree angles. The player must get the AI to crash into their wall while steering clear of theirs. *Armagetron Advanced* builds on that and gives the old-school arcade game a 21st century makeover.

You can grab the game's AppImage from its website and make it an executable either from the right-click context menu or with `chmod +x`. You can also install the game from Flathub with `flatpak install flathub org.armagetronad.ArmagetronAdvanced`.

Armagetron Advanced begins with a standard standalone game to help familiarise you with the gameplay, which basically remains the same. Your objective is still to box in your opponents with your trail, while making sure you don't smash into theirs or even yours. You can dive into the game's configuration and tweak critical elements such as the number of AI players, and define their IQ, the length of the wall, the size of the arena and other elements.

The *Advanced* bit in the name is the additional gameplay elements and modes. While you can still play against AI players on a local machine, the game is all about network play. What's unique about it is that the gameplay varies from one server to the next.

For instance, typically you can only change the direction of your bike by turning 90-degrees. However, some servers allow you to make 45-degree turns as well. Others have tweaked the size of the arena or tinkered with the physics, so your bike begins to accelerate when it's close to a wall.



```
bodhi left the game.
bodhi entered the game.
Creating grid...
Notepad entered the game.
Resetting scores...
Go (round 1 of 10)!
Press <v> or <c> to switch camera modes.
```

You can also host Armagetron Advanced on your local network and play with as many as 16 of your buddies.

BULLET HELL

Taisei Project

Version: 1.4.1
Web: https://taisei-project.org/news

Touhou is a popular shoot-'em-up game series known for its bullet hell gameplay. The objective in these games, also known as manic shooters, is for the players to navigate their character through the screen, dodging the endless stream of bullets or projectiles fired by your opponents, even as you fire some of your own at them.

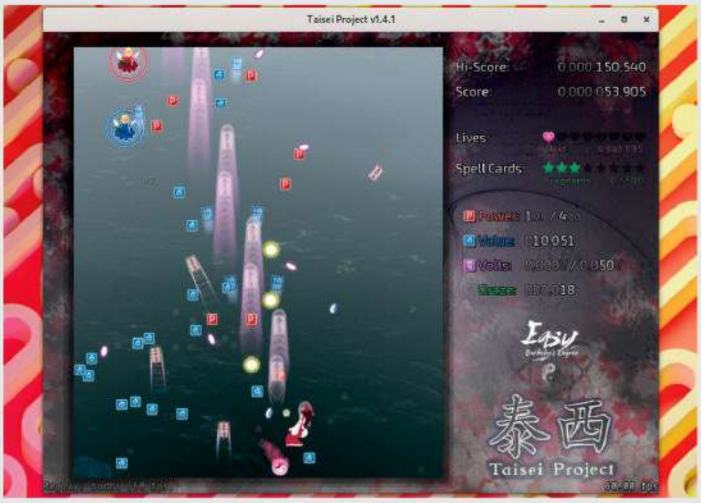
Taisei bills itself as a fan game of the *Touhou* series. The game is available on Flathub and can be installed with `flatpak install flathub org.taisei_project.Taisei`. The game embeds itself in your distro's application launcher when installed. You can also launch it from the terminal with `flatpak run org.taisei_project.Taisei`.

Before you begin the game, take a moment to review the control keys and perhaps even remap them.

The first time you start the game, you have to sit through its (rather long) backstory. Before the actual game, you first have to select from one of the four difficulty levels. If you haven't played manic shooters before, you're going to find even the easiest level a challenge. You then select from one of the three available characters, each of which has a different shot selection.

The objective is to dodge the projectiles, while you shoot at the enemies and collect powerups. If you are struck by an enemy projectile, you die. To make things easier, the game has shrunk your hitbox, so it is much smaller than your character.

The game has six stages. To progress, you have to fight each stage's boss and mid-boss, who fire projectiles that are different from the ones fired by the enemies you've dodged up until then. When you end a game, *Taisei* offers the option to save a replay, which you can watch later to enhance your gameplay.



If you are new to the bullet hell game, it's best to spend some time scrolling through Taisei Project's wiki to understand its gameplay and scoring system.

DOWNLOAD MANAGER

Varia

Version: 2024.3.20 **Web:** https://giantpinkrobots.github.io/varia/

Web browser downloaders are fairly robust. So much so that you don't need a download manager for most tasks. But if you're grabbing multiple large files, such as distro ISOs, there are benefits to a dedicated download manager like *Varia*.

The recommended way to install *Varia* is via Flathub:

```
flatpak install flathub io.github.giantpinkrobots.varia
```

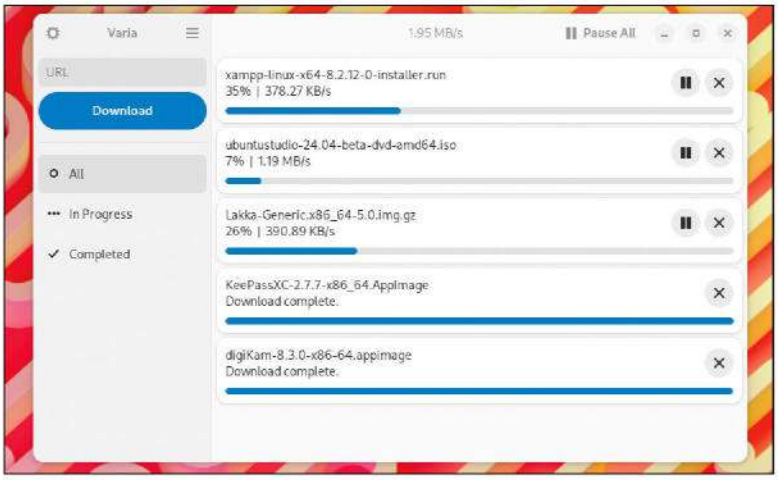
It is based on the venerable *Aria2*, which is why it has all the features you'd expect. You can use it to download files via the standard http and https servers, as well as via FTP, torrents and more.

In the top-left corner you have the URL box, where you paste the download links and hit the Download button to begin grabbing the file. Each download comes with controls to pause or remove them. You can use the sidebar to sort through the downloads that are in progress and those that have been completed.

The header bar lists the current download speed for all the files combined, along with a button to pause all downloads. Also, unlike the downloaders in your web browsers, you can use *Varia* to specify the number of simultaneous downloads, which helps optimise bandwidth use and download efficiency. By default, *Varia* does five concurrent downloads. Conversely, you can also ask *Varia* to limit the download speed, which helps you manage bandwidth usage according to your network constraints.

You can adjust some of its behaviour from the Preferences window, which you can bring up by clicking the gear button located in the top-left corner of the interface. While you're here, you can also change Download Destination and specify authentication details for downloads that need them.

The one missing feature in the app is the ability to set up scheduled downloads.



xampp-linux-x64-8.2.12-0-installer.run
35% | 378.27 KB/s

ubuntustudio-24.04-beta-dvd-amd64.iso
7% | 1.19 MB/s

Lakka-Generic.x86_64-5.0.img.gz
26% | 390.89 KB/s

KeePassXC-2.7.7-x86_64.AppImage
Download complete.

digiKam-8.3.0-x86-64.appimage
Download complete.

Varia also has browser extensions for Firefox and Chrome, which redirect all downloads from the browsers to Varia.

WEB BROWSER

Midori

Version: 11.3.1 **Web:** https://astian.org/midori-browser/

In its original avatar, *Midori* was the lightweight browser of choice for resource-conscious distros such as SliTaz, Bodhi Linux and Raspbian. It has since been acquired by Astian, which develops privacy-centric open source apps.

The cross-platform browser is available in several pre-compiled binaries. The best option is to grab the . **tar.bz2** compressed archive for your architecture. Once you have the file, fire up a terminal and type:

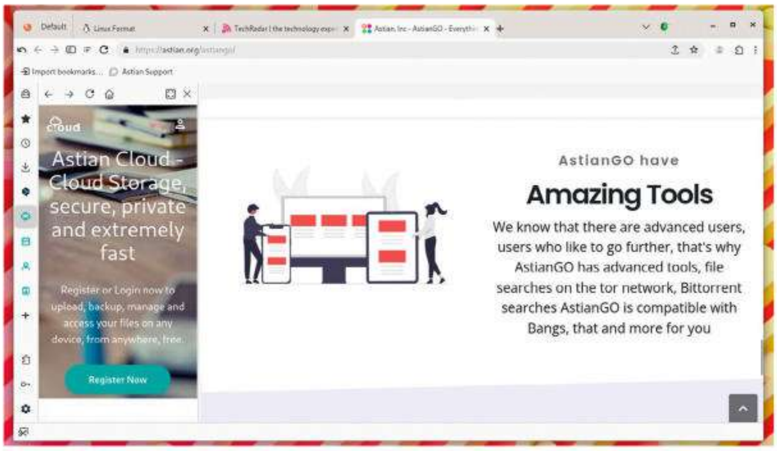
```
$ tar xf midori-11.3.1.linux-x86_64.tar.bz2
$ cd midori
$ ./midori
```

Midori uses *Firefox's* Gecko engine, but claims to have cut the bloat while adding new tools without compromising performance. One of the headline features of the browser are its customisation options. The developers claim these, along with the performance enhancers, help minimise the app's footprint on a laptop's battery.

You can hide or delete all of its toolbars. The browser also lets you customise them by rearranging their elements and moving them around the screen, as per your preference. Thanks to its Gecko base, *Midori* also supports all of *Firefox's* themes and extensions, and you can grab them from **http://addons.mozilla.org**.

Midori's main attraction, though, is its privacy features. Some of these are baked into the browser and some are implemented via popular extensions, such as uBlock Origin and Facebook Container, which are enabled by default.

Firefox has an add-on with which you can open private tabs in a normal browser session. *Midori* has baked this functionality into the browser itself. Another such out-of-the-box feature is the ability to organise tabs into workspaces.



Astian Cloud - Cloud Storage, secure, private and extremely fast

Register or Login now to upload, backup, manage and access your files on any device, from anywhere, free.

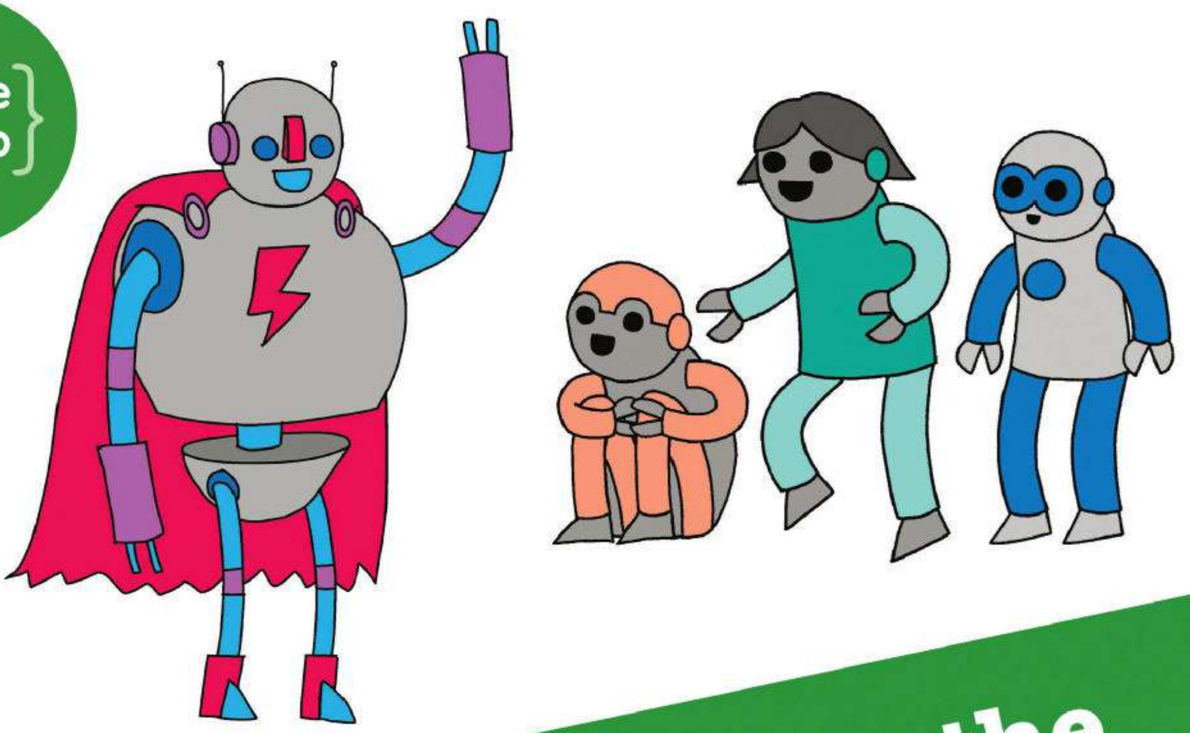
AstianGO have

Amazing Tools

We know that there are advanced users, users who like to go further, that's why AstianGO has advanced tools, file searches on the tor network, Bittorrent searches AstianGO is compatible with Bangs, that and more for you

Midori defaults to its privacy-respecting AstianGO search engine that claims to not track user activity, and doesn't display invasive adverts.

{code club}



Can you help inspire the next generation of coders?

Code Club is a nationwide network of volunteer-led after school clubs for children aged 9-11.

We're always looking for people with coding skills to volunteer to run a club at their local primary school, library or community centre for an hour a week.

You can team up with colleagues, a teacher will be there to support you and we provide all the materials you'll need to help get children excited about digital making.

There are loads of ways to get involved!

So to find out more, join us at **www.codeclub.org.uk**

PLAY-BY-MAIL

Part One! Don't miss next issue, subscribe on page 16!

Create turn-based internet games

David Bolton takes inspiration from his past and explains what postal games are, and what's needed to create and run one in the modern age.

OUR EXPERT

David Bolton worked for several years as a postal game developer back in the late '80s for what was then the UK's largest postal game company.

We're going to relive the heyday of the play-by-mail craze by creating our own multiplayer browser-based game. In this two-part series, we're looking at what you need to set up a game and providing an overview of the games processing. It's only called postal because that's how these games used to be run from the late '70s until the mid-'90s, when the internet came along. No stamps are needed now! Here is a list of the steps involved in setting up and running a game:

1. A game is developed then advertised.
2. Players sign up.
3. When enough players have joined, the game begins. The website is updated with the initial positions.
4. Players receive their initial positions.
5. Each turn, players submit their orders through a website or mobile app.
6. Once an hour (or whatever time period) all the game orders are processed.
7. Results are generated and the website/mobile app is updated.
8. Loop back to step 4 until the game finishes.

Those steps in more detail...

The type of game is entirely up to you. *Mafia*-type games are very popular. Take a look at the PBBG subreddit (**www.reddit.com/r/PBBG/**). PBBG stands for persistent browser-based games. Any form of multiplayer situation, such as space combat, military strategy games, sport – for example, football leagues – and, of course, fantasy games involving castles, dungeons and dragons are popular. Judging by that PBBG subreddit, there seems to be a new *Mafia* game coming out every week.

QUICK TIP

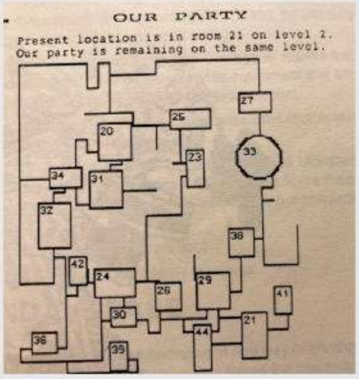
There's an old saying that you should always validate your inputs. Even if you wrote the part of the website or mobile app that accepts orders, you should still validate them in your program that reads and processes them. It's easy to make mistakes.

» WHAT IS PLAY-BY-MAIL (PBM)?

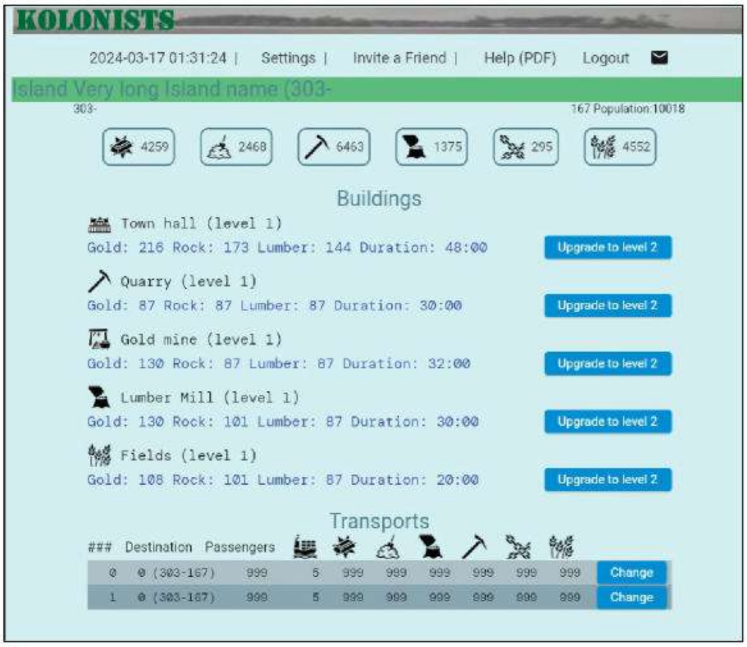
PBM is a form of commercial game playing that existed long before the web – from the '70s through to 1995. Games were advertised in magazines and players wrote in and asked to join the waiting list. When it was ready, the game would begin and players would be sent multipage printouts. Some games used special postcards, where players could enter their orders – usually simple commands – then posted orders to the company before the deadline. Each turn cost typically between 50p and £1.00.

At the games company, the orders were typed in, a program run and the turn was processed. One of the longest running games, *Quest* (**https://en.wikipedia.org/wiki/Quest_(game)**) was developed in 1989-1990 by the author while working at KJC Games and is still run today, with a major redesign in 2002. It was originally written in 40,000 lines of Turbo Pascal and took a year to develop. It was loosely based on the *Ultima* games, with up to 1,000 players whose parties of *D&D*-type characters travelled across a continent, went down dungeons and fought monsters and other parties.

Since the web appeared from 1995, games moved from postal to email and web, which saved postage and meant that turns could be done in hours, not days or weeks. The principle of submitting turns, having them processed and returned still works. Wikipedia (**https://bit.ly/lxf316wiki**) lists a number of active play-by-mail games but it's unlikely any are run by mail now!


OUR PARTY
Present location is in room 21 on level 2.
Our party is remaining on the same level.
27
26
20
23
33
34
31
32
38
42
24
26
29
41
30
21
44
36
39

A printed dungeon map from the original Quest play-by-mail game.



KOLONISTS

2024-03-17 01:31:24 | Settings | Invite a Friend | Help (PDF) Logout

Island Very long Island name (303-

303- 167 Population:10018

4259 2468 6463 1375 295 4552

Buildings

Town hall (level 1)

Gold: 216 Rock: 173 Lumber: 144 Duration: 48:00 Upgrade to level 2

Quarry (level 1)

Gold: 87 Rock: 87 Lumber: 87 Duration: 30:00 Upgrade to level 2

Gold mine (level 1)

Gold: 130 Rock: 87 Lumber: 87 Duration: 32:00 Upgrade to level 2

Lumber Mill (level 1)

Gold: 130 Rock: 101 Lumber: 87 Duration: 30:00 Upgrade to level 2

Fields (level 1)

Gold: 108 Rock: 101 Lumber: 87 Duration: 20:00 Upgrade to level 2

Transports

###	Destination	Passengers								
0	0 (303-167)	999	5	999	999	999	999	999	999	Change
1	0 (303-167)	999	5	999	999	999	999	999	999	Change

Screenshot of a Kolonists prototype.

The number of players can vary between a very small number, such as 10, and right up to 1,000 – it's entirely up to you, the game designer and programmer. But be aware that if you create a game that needs a very large number of players, it will take a long time to get enough players to start it.

Software requirements

First, you need to write a program to create the game. Now remember, you are creating not just a game but a system that can process multiple games. So, each game has its own data and storage area for orders and processed turn results. There's no right or wrong way to implement this – whether you use a database such as *SQLite* or *MySQL*, or maybe just flat files.

Quest was created using flat files; back in 1989, there wasn't much in the way of relational databases available, not without spending a large sum of money. *MySQL* didn't come along until 1995, while *SQLite* didn't appear until 2000.

There are three programs needed for the game:

1. For creating the initial game data.
2. For reading input and processing turns.
3. For generating output.

Typically, there are three sets of folders: **Input**, **Output** and **Data**. And below each is a numbered subfolder for each game.

The setup program creates a new subfolder for the game number and creates the games files in that folder – for example, **Data/1** for game 1 and so on. Another program processes the game data and creates the output files in **Output/1**.

At this point, we need to think about the game website. It serves two purposes. Firstly to advertise the game. Hint to game designers: put screenshots on your website, so we can see the game before signing up. Its second purpose is for turn entry.

The game's website

For the website, you need a domain name registered, hosting set up and an SSL certificate. As an example of a game website that has been created (although the game isn't ready to play just yet), take a look at **Turfwar.org**.

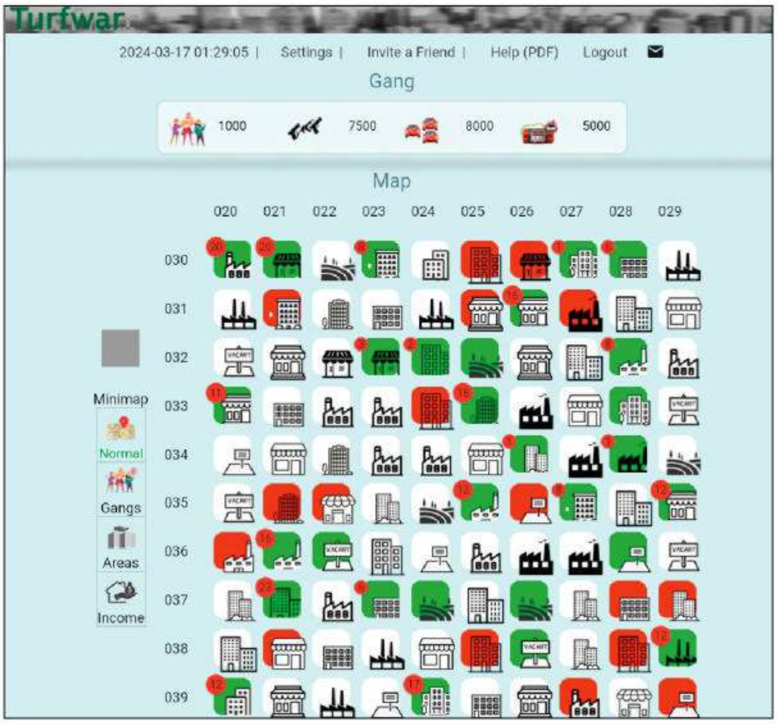
The domain costs around £20 per year, the hosting is on a server and costs around £5 a month, and the SSL certificate costs about £20 for four years.

When *Turfwar* is ready and working, there will be a login page, and players who have registered will be able to log in and then join a waiting list for a game. Once there are 75 players waiting, the game starts. The game output (which will be a PDF) will appear on the website after each turn for each player to download. They will then have an hour or more (depending on how often turns are processed) to enter their orders on a web page.

Once the deadline passes, all of the entered orders are passed to the VPS, which runs the game server software. If the website is on the same box that is hosting the game , it is just a simple copy to get the orders to the game server. Likewise, the PDF files output by the server have to be copied to the website. If they are on separate boxes, however, it's a little bit more involved, using *rsync*, for instance. Mobile apps can fetch the results via the website.

The game begins

Once enough players have joined, after running the program to generate the game data, the program to generate output is run and then the web server syncs with the game server and fetches the output



Here you can see an early gruesome prototype of a Turfwar city map.

QUICK TIP

Don't scrimp and do without an SSL certificate. Everybody expects them these days and browsers will not be happy if your site lacks one. Given that most hosting provides a means to generate a certificate request to send off and then install, and they cost £5 a year, they are a no-brainer.

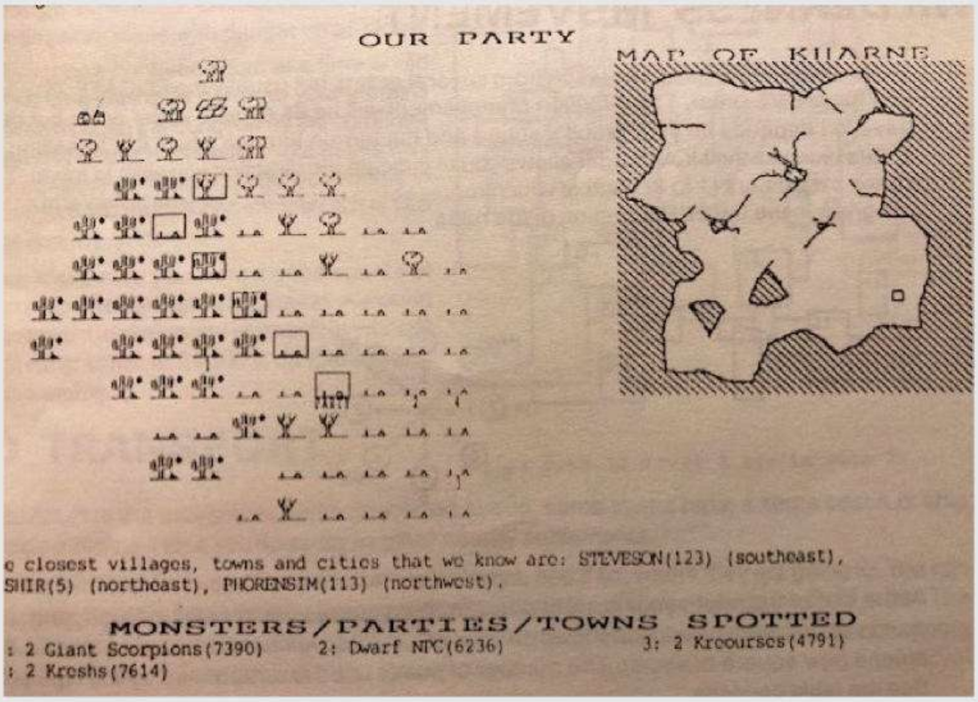
»

» WHAT DO YOU NEED?

You need an internet server box for doing game processing and a web server box for people to find out about the game, log in, enter orders and see the results of previous turns. If you want extra security, these can be separate boxes but so long as the server box can run a web server, you can get away with doing both on one server.

The great thing is you can get away with a cheap VPS running Linux, which can cost as little as a few pounds per month for basic hosting. It also means multiple games can be run on the same box. Processing a game turn for all players might take just a minute or so. If a turn is processed every hour or two, you can in theory run 50-100 games on the same server.

As for the language to use, this author's favourite is C#, which can create apps that run on a Linux box, but it's up to you. Other languages that do well include Java, Dart, FreePascal, D, Go or Rust, but any language is fine so long as it can create standalone apps, so it could also be in Python or Perl. As long as it has decent string handling, can read or write JSON files and has database support, any language that enables you to create console applications is good.



```
                    OUR PARTY                MAP OF KHARNE

e closest villages, towns and cities that we know are: STEVESON(123) (southeast),
SHIR(5) (northeast), PHORENSIM(113) (northwest).

        MONSTERS/PARTIES/TOWNS  SPOTTED
: 2 Giant Scorpions(7390)     2: Dwarf NPC(6236)        3: 2 Kreourses(4791)
: 2 Kroshs(7614)
```

A map from the original Quest PBM, printed on a dot-matrix printer.

files. Each player can now see their game position and start entering orders.

Orders, orders

What are orders? They're part of the game. For instance, in a *Mafia*-type game, you have a number of possible actions. Here are just a few.

- Recruit new soldiers.
- Take over a city block.
- Set up businesses in your blocks to provide income.
- Bribe law enforcement to look the other way.
- Attack an enemy-owned block.

The website or app lets you issue these commands. You have so many soldiers available to take actions, so a typical order, like Recruit, might be **R**,**5**,**64**, where **R** is short for Recruit, **5** is the number of soldiers you send, and **64** is the block number where you are recruiting from. Each time you add an order, it gets added to a list. Eventually, you've issued all your orders and you click the button to submit them. They get sent as a file (JSON, text and so on) to the web server and are put in the correct game input folder, ready to be processed.

The game results show things like how many soldiers you have in each block, what businesses are there and income they provide, and a map of the city blocks you own and nearby blocks.

Game processing

The game speed is set by how often turns are processed. Once an hour is fast and probably needs very dedicated players, while three hours per turn is a bit more manageable. You can have multiple games starting with different turn speeds.

The game processing program is the most important one; it reads the input files from the input folders for all players. It then processes each set of orders. There's a number of things that you should be aware of:

1. Make a backup of the game data before processing; both the current game files and the input files. If there's a bug and it crashes, you want to fix the bug, restore the game data and rerun it. No matter how much you test it, somebody will probably find a way to crash the game processing, so be prepared! With *SQLite* or flat files, backing up and restoring is just copying files. With a database server, it's a little bit more complicated.
2. You need to log all the actions and the results – good or bad – for each player. This can be as simple as appending to a text file for each player involved in an action. The idea is that when you generate the results PDF, it includes a 'here's what happened' along with any enemy actions, such as attacking your blocks, hitting your businesses, soldiers killed and so on.
3. Process the turn in multiple steps. First step: withdraw all the soldiers who are going on missions. They can't be used more than once per turn, so if they're out on a mission, they're not going to be there to defend when the block is attacked. Step

QUICK TIP

Once you've got the game programming underway, look to the PBBG subreddit and see if you can get game testers and people who have played other PBM games, and can provide feedback. Get them on your side and you'll get some of the best advocates and fans.

two: do all the attacks and so on. Step three: return the surviving soldiers to blocks after the missions. Step four: calculate income from all blocks you own – protection money and so on.

4. After it's all done, back up the game again.

Player updates

With all the game data after processing, plus text files from the 'action', the output files can be generated. This could be output into an HTML page so the player can view it on the website, or a PDF for a really nice formatted output.

If you are able to create mobile apps, then that might be a way to attract players, but you have to figure out how to fetch the results into a mobile app and on the website.

Thoughts on programming

If we go back to the Recruit, 5, 64 order, what is the upper limit for the number of soldiers and the block number? You have to define the range of every numeric parameter and have them all checked. It costs almost no processing time.

In the next article, we'll look at the website I/O and such things as the format of the input files.

The dating site problem

If you create a new dating website, you are going to have a similar problem as new games. There's nobody there and that can discourage people from joining. Now, a mate of ours who worked on a dating website told us that they added fake profiles, so that new signups would see people there and join up. The same thing can be done with these games. Unlike the fake profiles on dating websites, the fake players can be given a modicum of intelligence, so their positions are active. These fake players are computer-controlled.

The three phases of games

1. Signup period. This finishes when there are, say, 75% of the maximum players ready and the game starts.
2. Three or four months when players can still sign up.
3. Proper gameplay, with no new players allowed.

By using fake players at first, you can speed up phase one, and as new players sign up, just remove a fake player and either give the new player a position that has been played a bit or a brand new one. Players starting after a game has begun can suffer from being late starters, of course.

Monetising the game

There are different ways to do this and you may not need it – you could just let players play for free. But there are costs for even a cheap VPS, plus website hosting, domain fee and the SSL certificate fee. At the minimum, it's probably going to cost £250 a year and perhaps more, depending on the VPS costs. Free games attract a lot more players and you should expect to convert a small proportion of these to paying. Some of the monetisation methods are listed below.

A fixed annual fee, such as £20, if you provide extra features to players, such as using their own graphics for flags, running clans (alliances and so on), not showing adverts, or providing access to an API, so they can enter orders directly. What you must not do is make the game easier to win for paying players; just give them extras. You could also limit free players to play one game at a time, while paying players can play in three or more at the same time. For this, you have to put in anti-cheat methods to detect free players playing multiple games by using different emails.

Running adverts isn't considered too great an earner, because lots of people run ad blockers.

If you have a lot of games and players, it might be possible, however, to get advertisers to sponsor bonus cards. These are playable once and give players features – for example, play a 'Double recruitment this turn' bonus card. What cards are made available depends on your game, and the sponsors get a mention when every player in a game receives a bonus card. It's a form of advertising – you'd mention the sponsor and include a link to their website in the log file, so it shows up in their results.

You might allow players to buy a set of bonus cards for a couple of quid. This can open a real rabbit hole about players paying for things, and you need to balance it so players can't pay to win by buying lots. Players are not stupid and any hint of a game being pay-to-win will put people off. We'll discuss that and in-game currencies and dual-currency systems in the second tutorial, next month. LXF

QUICK TIP

Build in various mechanisms for communicating with players. This can be as simple as showing the numbers on the waiting list or the ability to send announcements to all players in a game or individual players who ask questions.

237th Armor, Hits: 0/2, Rng: 2, Ord: None

The Empire Deluxe wargame – played by email.

» GET OUR PLAY-BY-MAIL GAME… Subscribe now at http://bit.ly/LinuxFormat

CLASSIC DEMOS

Code a classic Star Wars scroller

Part Two! Don't miss next issue, subscribe on page 16!

There's nothing more classic than Star Wars, so **Ferenc Deák** brushes off his ancient dusty demo code for a scroller celebration.

OUR EXPERT

Ferenc Deák never throws anything away, which is why he still has all his demoscene files.

QUICK TIP

You can find and clone all the code from the GitHub page: **https://github.com/fritzone/lxf-demologia**

Last month, we had a short introduction to SDL2, and created a more-or-less usable coding framework on which we can base our implementation of some of the most notorious effects of the '90s demoscene. This month, we are continuing the recreation of those effects, and introducing some auxiliary mobility into the not-so-dormant screen we created last time.

The fire effect was one of those quick-and-dirty effects of the '90s that seized the attention of spectators at a demo party and ensured instant recognition from your peers. Implementing it involves creating a visual simulation of flames on a computer screen. This is typically achieved by representing the fire as a grid of cells, where each cell corresponds to a pixel and holds a value representing the intensity or temperature of the fire at that location. The process begins by initialising the grid, setting up the initial state of the fire, then iteratively updating the values of each cell based on the value of its neighbours. Randomness is often introduced to add a natural and organic appearance, and the intensity values are mapped to a colour palette set up to recreate the colour scheme of those found in real flames. The simulation is animated by continuously updating and rendering the grid.

The fire palette

Due to the nature of the effect, the palette used must be set up in such a way that colours at low indexes are darker (due to them being constantly averaged out, they simulate the extinction of the flame and the death of the fire), while colours at high indexes are bright, representing the new root of the fire, where the flames have the highest intensity. To achieve this effect, we have created the **generateFirePalette** method, which generates a corresponding set of red, green, blue series, and sets the colours correspondingly.

The actual implementation of the fire effect is in the following code sequence:

```
void updateScreen(Uint8* screen) {
 for (int x = XMIN; x <= XMAX; ++x) screen[YMAX *
SCREENSIZE_X + x] = rand() % 255;
 for (int x = XMIN; x <= XMAX; x++)
  for (int y = YMIN; y < YMAX; y++) {
   int total = 0;
   int divc = 1;
   total += getPixel(x-1, y + 1, screen);
   if(rand() % 2) { total += getPixel(x, y+1, screen); divc ++; }
   if(rand() % 2) { total += getPixel(x+1, y+1, screen); divc ++; }
   if(rand() % 2) { total += getPixel(x-1, y-1, screen); divc ++; }
   if(rand() % 2) { total += getPixel(x, y-1, screen); divc ++; }
   if(rand() % 2) { total += getPixel(x+1, y-1, screen); divc ++; }
   if(rand() % 2) { total += getPixel(x-1, y, screen); divc ++; }
   if(rand() % 2) { total += getPixel(x+1, y, screen); divc ++; }
   Uint8 a = static_cast<Uint8>( total / divc );
   putPixel(x, y, a, screen);
   if(rand() % 2) putPixel(x-1, y, a, screen);
   if(rand() % 2) putPixel(x+1, y, a, screen);
   if(rand() % 2) putPixel(x,y-1, a, screen);
   if(rand() % 256 == 15 ) {
     int rx = x - rand() % SCREENSIZE_X;
     int ry = y - rand() % SCREENSIZE_Y  ;
     Uint8 colAt = getPixel(rx, ry, screen);
     if(colAt >= 16) putPixel(rx, ry, rand() % 255, screen);}}}
```

Ugly and long as it is, it contains the entire implementation of the fire routine. Let's break it down, section by section. In the first loop, the bottom row of the screen is initialised to a random value between 0 and 255. This will be the new set of values for the fire algorithm that has the tendency to consume the pixels on the screen, thus needs constant feeding.

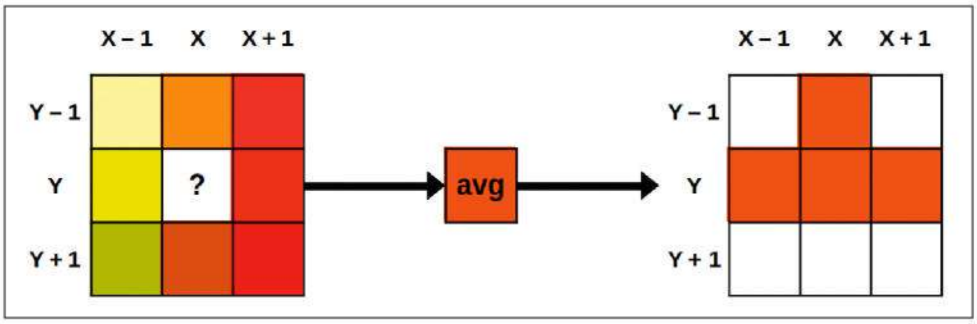
Then we traverse the screen, row by column, and for each pixel we calculate the new value of the current pixel as the average of the pixels surrounding it. Then we set the current pixel to be the newly calculated average value. The sequence `if(rand() % 256 == 15 )` is responsible for adding extra sparkles. The pixel-consuming algorithm will quickly devour those pixels and offer an interesting lively feeling to the fire.

In order to get a fire that resembles the real deal as much as possible, we need to call in extra help, in the form of a cellular automaton, called Conway's *Game of Life (see boxout, page 97 also **LXF284**)*.

In *Game of Life*, gliders and spaceships are interesting patterns that dynamically traverse the grid, exemplifying the complexity of the cellular automaton. Gliders are small, self-propelling entities, that move diagonally while leaving a distinct trail, while spaceships, larger in scale, can travel horizontally, vertically or diagonally, exhibiting diverse shapes and periodic movements. These dynamic patterns give rise to intricate and self-sustaining structures that add depth and interest to the evolving patterns on the grid, and we will be combining the gliders and spaceships with the pixel-consuming algorithm of the fire algorithm, achieving an effect where flames shoot out of the base fire, and this we call **fire ultimate**!

The final code for this effect is quite long, so we're printing only the bit that is specific to the *Game of Life* algorithm – the rest is almost identical to the original fire routine. Feel free to consult the full source code on the project's GitHub page *(see Quick Tip, left),* and please note that the code is a bit more complex, with a lot of action depending on random values; because fire is not something predictable, a lot happens without being able to know where the flames will go, and we wanted to emulate that as much as possible.

```
for (int x = XMIN; x < XMAX; ++x) {
 for (int y = YMIN + 1; y < YMAX; ++y) {
  int neighbours = (screen[(y - 1) * SCREENSIZE_X + x] > CONWAY_DIFFERENTIATOR ? 0 : 1) +
      (screen[(y + 1) * SCREENSIZE_X + x] > CONWAY_DIFFERENTIATOR ? 0 : 1) +
      (screen[y * SCREENSIZE_X + (x - 1)] > CONWAY_DIFFERENTIATOR ? 0 : 1) +
      (screen[y * SCREENSIZE_X + (x + 1)] > CONWAY_DIFFERENTIATOR ? 0 : 1) +
      (screen[(y - 1) * SCREENSIZE_X + (x - 1)] > CONWAY_DIFFERENTIATOR ? 0 : 1) +
      (screen[(y - 1) * SCREENSIZE_X + (x + 1)] > CONWAY_DIFFERENTIATOR ? 0 : 1) +
      (screen[(y + 1) * SCREENSIZE_X + (x - 1)] > CONWAY_DIFFERENTIATOR ? 0 : 1) +
      (screen[(y + 1) * SCREENSIZE_X + (x + 1)] > CONWAY_DIFFERENTIATOR ? 0 : 1);
   if (screen[y * SCREENSIZE_X + x] < CONWAY_DIFFERENTIATOR) // Cell is alive
   {
    if (neighbours < 2 || neighbours > 3) {
     int total = screen[(y + 1) * SCREENSIZE_X + (x - 1)] + screen[(y + 1) * SCREENSIZE_X + x] + screen[(y + 1) * SCREENSIZE_X + (x + 1)] + screen[y * SCREENSIZE_X + (x - 1)] + screen[y * SCREENSIZE_X + x] + screen[y * SCREENSIZE_X + (x + 1)];
     screen[y * SCREENSIZE_X + x] = static_cast<Uint8>(total / 7);  // Cell dies
    }
   }
   else { // Cell is dead
    if (neighbours == 3) {
     screen[y * SCREENSIZE_X + x] = 255;  // Cell becomes alive
    }
   }
```



The main idea behind the blur algorithm used in the fire effect.

This code is a modification of Conway's *Game of Life* algorithm, adapted to our fire routine, where we introduce a different rule set for the evolution of cells on the grid. We treat the entire fire display as the grid for the automaton, and the key differences from the original algorithm involve the handling of live and dead cells based on the number of live neighbours using the value **CONWAY_DIFFERENTIATOR**, which serves as a threshold to distinguish between live and dead cells. Let's break down the code:

1. Neighbourhood calculation

- The code iterates through each cell in the specified range (**XMIN** to **XMAX** for x, and **YMIN + 1** to **YMAX** for y).
- For each cell, it calculates the number of live neighbours by checking the values of neighbouring cells in the **screen** array. The condition **screen[...] > CONWAY_DIFFERENTIATOR** determines whether a cell is considered alive.

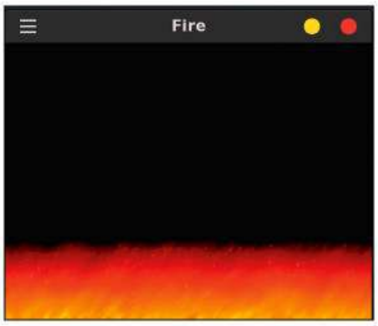
» THE HUMBLE TOOLS PROGRAMMER

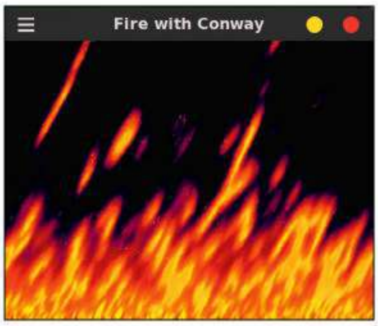
Just as not all that glitters is gold, in the demoscene, not all who code write low-level assembly and hexa codes. There is a great need for programmers to write quick and dirty tools, such as for transforming an image into a format that can be loaded easily into the current compo, or another that writes a piece of code that generates various other data to be used for different purposes.

Our scrolling *Star Wars* demo needs such a tool to transform the text that scrolls up the screen into a format that's easily recognised and loaded into the demo. For this we wrote a small C++ program that loads a PNG file (using the freely available *LodePNG*), scales it to a specific size (using bilinear filtering), performs a palette reduction to 256 colours (using the k-means clustering algorithm), and saves the result into a file in the following format: firstly, we want it to be a text file, to easily visualise the data structures behind the scenes of a scene compo. The first line is the size of the image, such as: 640x400 . We pick a size that fits the resolution of the screen we are going to work with for easy handling.

The following lines are all four bytes, each representing a palette entry for the image. Then we have the image data, one byte for each pixel. We know this is not the best format, but it exemplifies how the process works at a level that a binary file is unable to reproduce. The source of the demo is then complemented with a new function that loads an image from a specific custom file and deposits it in an image array, to be used at later stages.

»





Above: Our flame effect in action, but we can make it more realistic.

Above-right: There's a theory that fire is actually alive, but this certainly isn't.

2. Updating live cells

- If the current cell is alive (`screen[y * SCREENSIZE_X + x] < CONWAY_DIFFERENTIATOR`), it checks the number of live neighbours.
- If the number of live neighbours is less than 2 or greater than 3, indicating under- or overpopulation, a new value is calculated based on the average of certain neighbours, and the current cell is updated to the new value. This simulates the death of the cell.

3. Updating dead cells

- If the current cell is dead (`screen[y * SCREENSIZE_X + x] >= CONWAY_DIFFERENTIATOR`), it checks the number of live neighbours.
- If the number of live neighbours is exactly 3, it updates the current dead cell to become alive by setting its value to 255.

The true beauty of these effects lies in the simplicity of their rules, which, when tweaked or expanded upon, can lead to incredibly diverse and unpredictable patterns. So, dive into the code, experiment with different parameter values, and tinker with the rules, because these experiments can reveal an interesting interplay between order and chaos, and produce something fascinating to share with us.

I like to move it, move it

Our very first demo effect we saw in the introduction to SDL last month was where we convinced an otherwise static red square to move along the screen, or using scene terms, we implemented scrolling. There was a lot of scrolling in the demos of the '90s, but it all shares one common factor: something moves on the screen. Whether it's vertical, horizontal, diagonal, running along the points of a curve, distorted or not, something is moving.

In the '90s, these effects were achieved by overabusing the existing hardware, sometimes using undocumented features, which worked on one video card and not on another, and mostly involved very low-level assembly code, direct hardware access and lots of luck. We will not venture into these dangerous realms, but will create something that is visually intriguing and constitutes a medium-difficulty effect.

To exemplify scrolling in code, we chose an interesting effect: the *Star Wars* scroll. Everyone who has seen at least one episode of the franchise has experienced it first-hand: in a galaxy far, far away, there was a long text describing the heroic efforts of a handful of rebels fighting against the evil empire.

Implementing it, however, is not that difficult. One of the possible implementations is to have the text printed on a back buffer, and from there we scale in line by line to the front buffer, achieving the required distortion effect, giving the hint of 3D text disappearing slowly in the distance.

QUICK TIP

There is an excellent book covering the origins of the demoscene: *Freax* by Tamás Polgár. It is recommended for anyone wanting to dig deeper and find out how this fascinating subculture started.

The following C++ function is responsible for this:

```
std::vector<uint8_t> scaleArray(const uint8_t*
inputArray, size_t originalLength, double percentage) {
  if (percentage < 0.0 || percentage > 100.0)
    return std::vector<uint8_t>(inputArray, inputArray
+ originalLength);
  size_t newLength = static_cast<size_t>(originalLength
* (percentage / 100.0));
  std::vector<uint8_t> scaledArray(newLength);
  double step = static_cast<double>(originalLength - 1)
/ static_cast<double>(newLength - 1);
  for (size_t i = 0; i < newLength; ++i) {
    double index = i * step;
    size_t lowIndex = static_cast<size_t>(index);
    if(lowIndex > originalLength) lowIndex = 0;
    size_t highIndex = std::min(lowIndex + 1,
originalLength - 1);
    double fraction = index - lowIndex;
    scaledArray[i] = static_cast<uint8_t>((1.0 - fraction)
* inputArray[lowIndex] + fraction *
inputArray[highIndex]);
  }
  return scaledArray;
}
```

The function takes an array and scales it based on a percentage, performing linear interpolation to determine the values of the scaled array.

Linear interpolation is a method for estimating values between two known values. In the context of this function, linear scaling is used to interpolate values between adjacent elements in the input array to determine the values for the scaled array.

Let's denote the two adjacent elements in the input array as **inputArray[lowIndex]** and **inputArray[highIndex]**. The linear interpolation formula between these two elements can be written as:

```
interpolatedValue=(1–fraction)×inputArray[lowIndex]+
fraction×inputArray[highIndex]
```

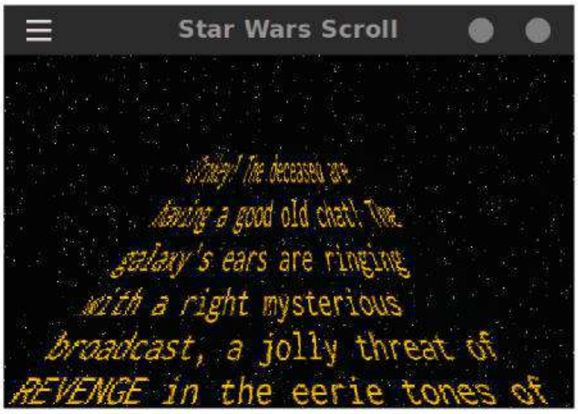
Here, **fraction** is a value between 0 and 1, representing the proportion of the distance between **lowIndex** and **highIndex**, and **interpolatedValue** is the value between **inputArray[lowIndex]** and **inputArray[highIndex]** based on the fraction.

This formula calculates the value at a given index in the scaled array by blending the values of the adjacent elements in the original array based on their distances from the index.

For the scroll effect, the text is loaded from our custom image format *(see boxout, previous page)* into a buffer, and to bootstrap the scroll effect, we need to do some initialisations, such as:

```
int currentRow = YMAX - 1;
int textureEndRow = 1;
static Uint8 row[SCREENSIZE_X] = {0};
```

The variables have the following meanings: **currentRow** is the row on the screen where we start drawing the scaled texture with the infamous words scrolling up the screen. The **textureEndRow** variable tracks the last row of the texture that is to be drawn, which is usually drawn full-width – 100%. Then the **row** variable is just a helper variable that is used to draw the



ney! The deceased are having a good old chat! The galaxy's ears are ringing with a right mysterious broadcast, a jolly threat of REVENGE in the eerie tones of

If you're not flashing back to your Amiga days, you never had any!

currently scaled row on to the proper place in the screen buffer.

After these have been successfully set up, we can move to the next phase and do the actual drawing on the screen with the following piece of code:

```
double beginScale = 100.0 - static_cast<double>(textureEndRow)/4.0  + 1.0;
for(int cr=0; cr<=textureEndRow; cr++)
{
 memset(row, 0, SCREENSIZE_X);
 if(beginScale < 0) beginScale = 0;
 auto t = scaleArray(textBuffer + screenWidth * cr, screenWidth, beginScale);
 for(size_t j=0; j<t.size(); j++) row[SCREENSIZE_X / 2 - t.size()/2 + j] = t[j];
 memcpy(screen + currentRow * screenWidth + screenWidth * cr, row, SCREENSIZE_X);
 if(cr % 4 == 0) beginScale += 1.0;
}
textureEndRow ++;
currentRow --;
```

The lines above are part of the main loop and are responsible for rendering scaled the texture of the text buffer on to a screen buffer. This rendering is done in a **for** loop, where **cr** represents the current row, with its value starting from the first line of the texture to the current last row of the texture, tracked by the variable **textureEndRow**. The variable **textureEndRow** is responsible for keeping track of the last row of the text buffer that is to be drawn on the screen. As you can see, with each iteration this value is increased, so more and more of the texture is drawn in each step.

The variable **beginScale** keeps the scale value for each row of the text texture. It starts with a value that the first row should be scaled to, then is increased for every row according to the size of the texture and the desired scaling ratio.

The actual scaling of the current row of the text texture is done in the line `auto t = scaleArray(textBuffer + screenWidth * cr, screenWidth, beginScale);`. Then the work buffer held in the **row** variable is updated in a way that the scaled **t** buffer is drawn symmetrically in the centre of it to reasonably resemble the effect of the text going upwards in a 3D manner.

Finally, the **row** buffer is copied to the correct location in the **screen** buffer, and we can move on to the next row in the texture after adjusting the scaling value for the rows to be processed.

When all the rows that have to be processed in the current iteration have been processed, we can move to the next row by increasing the **textureEndRow** variable and signal to the screen to decrease the value of the row that will hold the output of the texture.

Once done, and we've added a bonus star field, we get the final result *(see screenshot, above-left).*

It's quite pixellated, but still better than the ones we made in the '90s, because here we could use a bigger resolution than 320x200. If you want to tinker with the code, the two values of **4** were not chosen randomly, so feel free to experiment and see what happens.

A peek at the future

We've only scratched the surface of the mathematical intricacies involved. Next time, we are plunging into the more advanced domains of trigonometry, functions and other complex mathematical concepts.

» GAME OF LIFE

British mathematician John Conway's *Game of Life* from 1970 is a classic example of a cellular automaton – a mathematical model that consists of a grid of cells, each of which can be in one of a finite number of states. It is famous for its simplicity and the complexity that can arise from a few basic rules. Here are its key components:

1. Grid

- *Game of Life* is played on a two-dimensional grid of square cells.
- Each cell can be either alive or dead.

2. Neighbourhood

- Each cell has a neighbourhood, typically defined as its eight surrounding cells (horizontal, vertical, and diagonal).

3. Rules

- The state of a cell in the next generation is determined by its current state and the states of its neighbours.
- The rules are simple:
  Birth A dead cell with exactly three live neighbours becomes alive in the next generation.
  Survival A live cell with two or three live neighbours survives to the next generation.
  Death In all other cases, a cell dies or remains dead in the next generation.

4. Iterations

- The game progresses in discrete steps, or generations.
- At each generation, the rules are applied simultaneously to all cells to determine their new states.

5. Patterns

- The initial configuration of live and dead cells is often referred to as the seed or pattern.
- *Game of Life* is known for the emergence of various interesting and complex patterns, such as oscillators, gliders, spaceships and stable structures.

The simplicity of the rules belies the complexity that can arise during the simulation. *Game of Life* has been used as a metaphor for emergent complexity in various fields, including computer science, biology and philosophy. It has also served as a recreational and educational tool for exploring concepts related to cellular automata and dynamic systems.

» BLAST YOUR BRAIN WITH MATHS! Subscribe now at http://bit.ly/LinuxFormat

NEXT MONTH

LXF317 will be on sale Tuesday 25th June 2024

SAVE YOUR OLD PCs!

Bring out your dead PCs! We'll use 32-bit distros and restoration tricks to revive your old hardware.

Get to grips with Flatpaks

Stop Snaps, punt packages, away with AppImages and switch to the best app system with Flatpaks.

Top terminals!

Everyone's happy with Bash but there are alternative command-line shells out there to try – and they're free!

Accelerating AI

Discover how to top out your GPU for AI optimisations and accelerating your machine-learning systems.

GIMP 3.0

It's slated for a May release, so we'll take a first look at the biggest release in years for the key graphical editor!

Content of future issues subject to change. We might have been taken to the tip…

CREDIT: Magictorch

LINUX FORMAT
The #1 open source mag

Future Publishing Limited, Quay House, The Ambury, Bath, BA1 1UA
Email contact@linuxformat.com

EDITORIAL
Editor-in-chief Neil Mohr
Art editor Fraser McDermott
Production editor Katharine Davies
Group editor-in-chief Graham Barlow
Group art director Warren Brown
Editorial contributors
Mike Bedford, Denise Bertacchi, Jonni Bidwell, David Bolton, Neil Bothwick, Stuart Burns, James Capell, Christian Cawley, Krishi Chowdhary, Ferenc Deák, Nate Drake, Matt Holder, Bram Lodewijks, Jon Masters, Nick Peers, Les Pounder, Michael Reed, Mayank Sharma, Shashank Sharma, Abbie Stone, Ganesh TS
Cover illustration Magictorch.com
Raspberry Pi is a trademark of the Raspberry Pi Foundation.
Tux credit: Larry Ewing (lewing@isc.tamu.edu).
Content production Adequate Media Limited

ADVERTISING
Commercial sales director Clare Dove
clare.dove@futurenet.com
Advertising director Lara Jaggon
lara.jaggon@futurenet.com
Account director Andrew Tilbury
andrew.tilbury@futurenet.com

INTERNATIONAL LICENSING
Head of print licensing Rachel Shaw
Linux Format is available for licensing and syndication.
To find our more contact us at licensing@futurenet.com
or view our content at www.futurecontenthub.com

NEW SUBSCRIPTIONS & BACK ISSUES
Web www.magazinesdirect.com
UK 0330 333 1113 **World** +44 (0) 330 333 1113

EXISTING SUBSCRIPTIONS
Web www.mymagazine.co.uk
UK 0330 333 4333 **World** +44 (0) 330 333 4333
Subscription delays: Disruption remains within UK and international delivery networks. Please allow up to seven days before contacting us about a late delivery to **help@magazinesdirect.com**

CIRCULATION
Newstrade & retail category director Ben Oakden

PRODUCTION AND DISTRIBUTION
Group head of production Mark Constance
Production manager Nola Cokely
Senior ad production manager Jo Crosby
Digital editions manager Jason Hudson

THE MANAGEMENT
Managing director technology group Paul Newman
Global head of design Rodney Dive
Commercial finance director Tania Brunning
Printed by William Gibbons & Sons
Distributed by Marketforce UK
121-141 Westbourne Terrace, London W2 6JR. **www.marketforce.co.uk**
For enquiries email: **mfcommunications@futurenet.com**

Order and access back issues: If you are an active subscriber, you have instant access to back issues through your iOS or Android device/s. Your digital magazine entitlement is available at no additional cost and no further action is required. Pocketmags library may not have access to the full archive of digital back issues. You will only be able to access the digital back issues as long as you are an active subscriber.

To purchase single back issues (print format only): Visit www.magazinesdirect.com (click on Single Issues tab) or email help@magazinesdirect.com. For further help call +44(0)330 333 1113. Lines are open Mon- Fri 8.30am-7pm and Sat 10am-3pm UK time. Magazinesdirect.com is owned and operated by Future Publishing Limited.

BAR rates for Linux Format: £84.37 for UK, €171 for Europe, $194 for USA, £149 for rest of world.

Linux® is the registered trademark of Linus Torvalds in the US and other countries. GNU/Linux is abbreviated to Linux throughout for brevity. Where applicable, code printed in this magazine is licensed under the GNU GPL v2 or later. See www.gnu.org/copyleft/gpl.html. All copyrights and trademarks are recognised and respected.

Disclaimer: All contents © 2024 Future Publishing Limited or published under licence. All rights reserved. No part of this magazine may be used, stored, transmitted or reproduced in any way without the prior written permission of the publisher. Future Publishing Limited (company number 2008885) is registered in England and Wales. Registered office: Quay House, The Ambury, Bath BA1 1UA. All information contained in this publication is for information only and is, as far as we are aware, correct at the time of going to press. Future Publishing Limited cannot accept any responsibility for errors or inaccuracies in such information. You are advised to contact manufacturers and retailers directly with regard to the price of products/services referred to in this publication. Apps and websites mentioned in this publication are not under our control. We are not responsible for their contents or any other changes or updates to them. This magazine is fully independent and not affiliated in any way with the companies mentioned herein.

If you submit material to us, you warrant that you own the material and/or have the necessary rights/permissions to supply the material and you automatically grant Future Publishing Limited and its licensees a licence to publish your submission in whole or in part in any/all issues and/or editions of publications, in any format published worldwide and on associated websites, social media channels and associated products. Any material you submit is sent at your own risk and, although every care is taken, neither Future nor its employees, agents, subcontractors or licensees shall be liable for loss or damage. We assume all unsolicited material is for publication unless otherwise stated, and reserve the right to edit, amend and adapt all submissions. All contents in this magazine are used at your own risk. We accept no liability for any loss of data or damage to your systems, peripherals or software through the use of any guide.

We are committed to only using magazine paper derived from responsibly managed, certified forestry and chlorine-free manufacture. The paper in this magazine was sourced and produced from sustainable managed forests, conforming to strict environmental and socioeconomic standards.

Future is an award-winning international media group and leading digital business. We reach more than 57 million international consumers a month and create world-class content and advertising solutions for passionate consumers online, on tablet & smartphone and in print.

Widely Recycled

FUTURE
Connectors.
Creators.
Experience
Makers.

Future plc is a public
company quoted on the
London Stock Exchange
(symbol: FUTR)
www.futureplc.com

Chief Executive Officer **Jon Steinberg**
Non-Executive Chairman **Richard Huntingford**
Chief Financial and Strategy Officer **Penny Ladkin-Brand**

Tel +44 (0)1225 442 244

ipso. For press freedom with responsibility

The Electronic Frontier Foundation is the leading nonprofit organization defending civil liberties in the digital world. Founded in 1990, EFF champions user privacy, free expression, and innovation through impact litigation, policy analysis, grassroots activism, and technology development. We work to ensure that rights and freedoms are enhanced and protected as our use of technology grows.

EFF.ORG

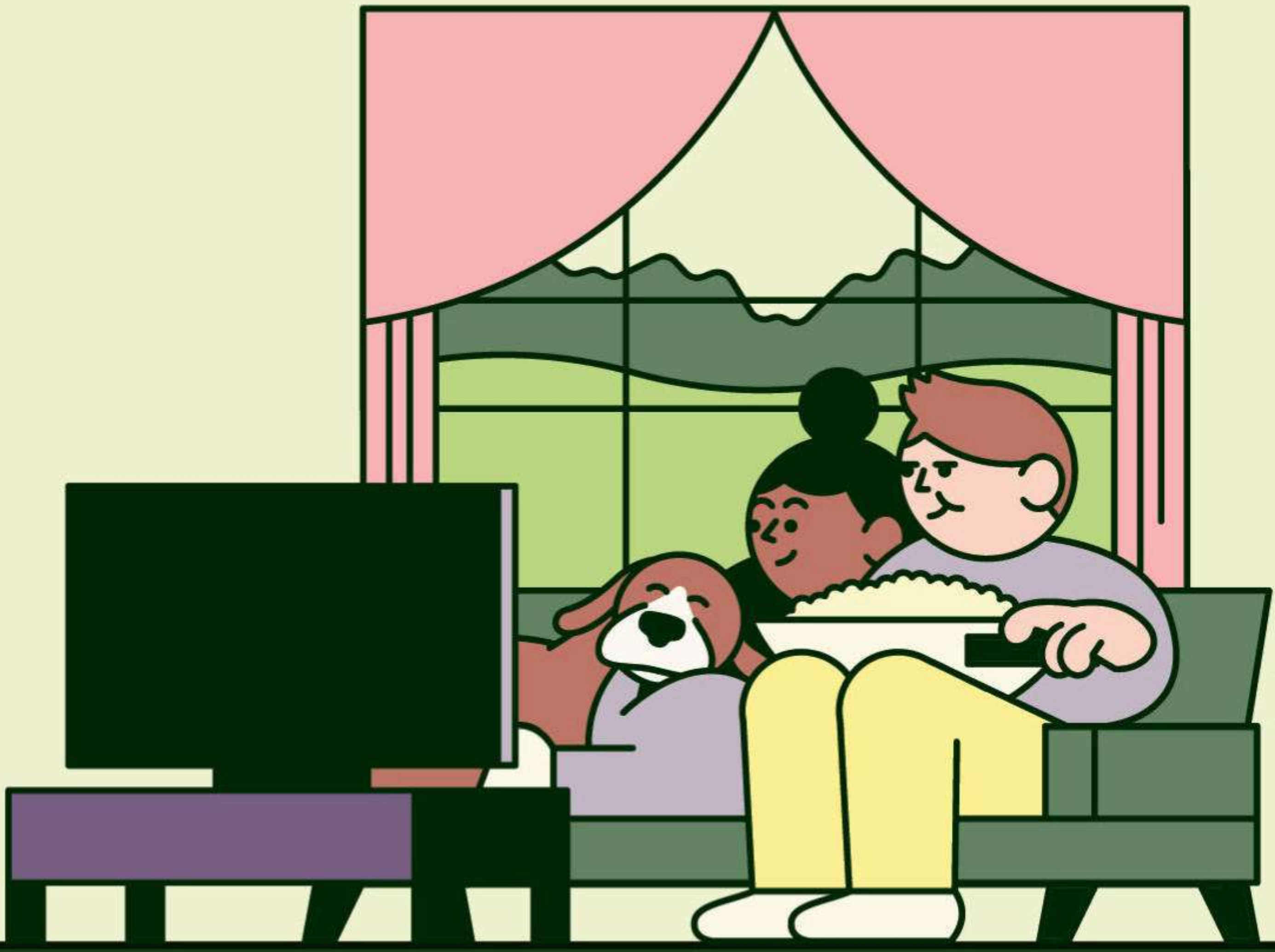
ELECTRONIC FRONTIER FOUNDATION

Protecting Rights and Promoting Freedom on the Electronic Frontier

Advertisement

SAVVY SAVERS, STEP THIS WAY.

Discover cover options for car, home and travel insurance, plus broadband and more.



GO. COMPARE

Get more information and compare quotes at Go.Compare.

9021

9000